03-60-510
Literature Review and Survey

A Survey:

Multi-Agent Coordination in the Meeting-Scheduling Problem

Hijaz Al-Ani
alani@uwindsor.ca

Instructor: Dr. Richard Frost
Supervisor: Dr.Ahmed Tawfik

1/27/2007

TABLE OF CONTENTS

ABSTRACT

Multi-Agent systems are being more and more widely used to address various distributed combinatorial real-world problems. One such problem is meeting scheduling (MS) that is characterized essentially by two features defined from both its inherently distributed and dynamic nature. In addition, in real world applications, users usually have conflicting preferences. However, scheduling a meeting is generally difficult in that it attempts to satisfy the preferences of all participants. All participants can agree to a schedule in which a portion of their preferences are not satisfied, since preferences are regarded in term of their relative day-life importance. This survey focuses on previous and current efforts to solve the meeting-scheduling problem based on distributed agents.

1 INTRODUCTION

Meeting Scheduling is a common task for organizations of any size. In simple terms, it involves searching for a time and place when and where all the meeting participants are free and available. There may be global organization and local individual constraints and preferences on the time and/or location.

Meeting Scheduling is a naturally distributed task (Ephrati et al., 1994; Garrido L and Sycara K, 1996; BenHassine et al., 2004), which is time-consuming, iterative, and somewhat tedious. It can take place between two persons or among several persons. Sometimes, people just try to schedule one meeting. However, most of the time people need to schedule many meetings at the same time taking into account several constraints.
Each potential attendee needs to take into account his/her own meeting preferences and calendar availability. Most of the time, each attendee has some uncertain and incomplete knowledge about the preferences and calendar of the other attendees; in fact, people usually try to keep their calendar and preference information private. During the meeting scheduling process, all attendees should consider the main group goal but they also take

into account individual goals (i.e. to satisfy their individual preferences). Solving the MS problem involves finding a compromise between all attendees requirements, usually conflicting, for meeting (i.e. date, time an duration). Thus, this problem is subject to several constraints, essentially related to availability, timetabling and preferences of each user. Automating meeting scheduling is important, therefore, not only because it can save time and effort on the parts of humans, but also because this may lead to more efficient schedules and to changes in how information is exchanged within organization. There exist several commercial product but they are just computational calendars with some special features, (Sen and Durfee, 1991; Garrido and Sycara, 1996; Tsuruta and Shintani, 2000; BenHassine et al., 2004) mentioned that non of these products is truly autonomous agent capable of communicating and negotiating with other agents in order to schedule meetings in a distributed way taking into account the users preferences and calendar availability.

In this survey we discuss research efforts in solving the meeting-scheduling problem and comparing different approaches and methods with focusing mainly on viewing meeting scheduling as a distributed task based on multi-agent systems. This survey is organized as follows. Section 2 introduces Agent and Multi-Agent system definitions. In section 3, we describe the meeting scheduling problem. Section 4 show past efforts solving MS problem using mutli-agent systems with summarizing different approaches. Finally, the conclusion is presented in section 5.

2 MULTI-AGENT SYSTEMS

Multi-Agent systems are systems composed of multiple interacting elements, known as agents. (Jennings, 2000; Wooldridge, 2002) define multi-agents system as the system that contains a number of agents, which interact with one another through communication. The agents are able to act in an environment; different agents have different ‘spheres of influence’, in the sense that they will have control over, or at least be able to influence, different parts of the environment.

Multi-agent systems are a relatively new sub field of Computer Science, they have only been studied since about 1980, and the field has only gained widespread recognition since about the mid 1990s. However, since then, international interest in the field has grown enormously. This rapid growth has been spurred at least in part by the belief that agents are an appropriate software paradigm through which to exploit the possibilities presented by massive open distributed systems.

2.1 Agent

An obvious way to start this survey would be by introducing a definition of the term agent. After all, this survey is based on multi-agent systems. Wooldridge mentioned in his book (Wooldridge, 2002) that there is no universally accepted definition of the term agent, but sort of definition is important. An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives. The definition presented here is adapted from (Wooldridge and Jennings, 1995)

3 MEETING-SCHEDULING PROBLEM

Meeting Scheduling is the process of determining a starting time and an ending time of an event in which several people will participate. Many requirements, constraints, of these participants must be taken into account in scheduling meeting (Tsuruta and Shintani, 2000).

The meeting-scheduling problem is a type of negotiation problem. A negotiation problem is associated with a set of fixed and variable attributes. The initiator of the meeting determines which attributes are fixed and which are variable. Those that are variable will be negotiated. For example, a person calling a meeting might tell his assistant: ‘‘I would like to a hold a project meeting sometime next week, preferably next Wednesday afternoon, with Tom’’ In this example, the type of meeting, desired time period when the meeting is to be held, and the attendees are fixed attributes, while the day and time are variable attributes. The following are examples of meeting attributes:

- Initiator: The host or initiator of the meeting. A person might consider a meeting called by his/her immediate supervisor to be more important the others, for example.
- Rank: The rank or position of the person calling the meeting. Values can be any rank or position within the organization, such as: CEO, CFO, CTO, VP R&D, VP Sales, etc.
- Attendees: The participants or invitees, a list of people that need to attend the meeting. This list may be further classified according to priorities of the attendees, such as those that ‘‘must’’, ‘‘should’’ or ‘‘can’’ attend the meeting. All those that ‘‘must attend,’’ must be all available before the meeting can be confirmed, otherwise it will be cancelled. ‘‘Should attend’’ are the normal participants of the meeting. ‘‘Can attend’’ are casual observers; their availability will not affect the meeting schedule.
- Type: The type of meeting. For example, values might include: general, departmental, group, strategic, inter-departmental, technical, marketing, sales, project, interview, etc. This value can be used to determine the priority of the meeting during scheduling. Higher-priority meetings might be scheduled earlier or might even take over timeslots from previously scheduled lower priority meetings, i.e., unschedule a meeting, which might get automatically rescheduled by the Meeting Agent that is looking after that meeting. Unscheduling is performed using conflict resolution.
- Period: The time period that the meeting should be held, such as ‘‘within the coming 2 weeks’’, ‘‘within this week’’, ‘‘within Friday’’, etc. The exact date and time is represented by other attributes.
- Duration: The length of the meeting. Values may be a number of hours or minutes.
- Part-of-day: The part of the day that the meeting will be held. For example, values may be from: breakfast, morning, lunch, afternoon, dinner or evening. This is a coarser grain classification than hours and seems to be more nature in defining time preferences.

4 SURVEY: BASIC MEETING-SCHEDULING SOLVERS APPROACHES

Many research efforts dealing with Meeting Scheduling (MS) problems have been proposed in the literature (BenHassine et al., 2004; Franzin et al., 2004; Modi and Veloso, 2004; Freuder et al., 2001; Tsuruta and Shintani, 2000; Luo et al., 2000; Sen et al., 1997; and Ephrati et al., 1994). However, recently multi-agent systems were widely used to address many real-world combinatorial applications. The main reason for using an agent-based approach to solve MS problems is that agents can accomplish their tasks through cooperation while allowing the user to keep their privacies (BenHassine, 2005). In this section, we discuss the main mechanisms approaches based on multi-agent system to solve the MS problem.

4.1 The Partial-Constraints-Satisfaction Problem (PCSP) approach

The first multi-agent approach to MS problems using partial CSP was introduced by (Freuder and Wallace, 1992).

The second work (Lemaitre and Verfaillie, 1997) used Distributed-Valued Constraint-Satisfaction Problem DVCSP to formalize MS problems, this work propose a formalization of DVCSP and a greedy repair distributed algorithm for solving DVCSP, in this algorithm during an agent turn, other agents must not change their local assignment. The third work (Tsuruta and Shintani, 2000) applied a distributed synchronous algorithm to MS problems formalized as DVCSP and some agents can change their local assignments simultaneously. The problem addressed by the authors is that meeting scheduling is over-constrained and no solution exists that can satisfy all constraints. They try to develop new method for scheduling meetings that satisfies as many of the important constraints as possible by formalizing MS problem as Distributes Valued Constraints Satisfaction. The idea of this algorithm as presented is an agent corresponds to each group member, this agent maintains its user calendar and preferences for meetings and acts on behalf of its user in meeting scheduling; users are able to keep information regarding their calendars and preferences private.

From their experiments, the authors claim that the proposed algorithm is cost-effective in comparison to the DOC method (Bakker et al., 1993), it can discover a semi-optimal solution to over constrained MS problem in practical time, and determining an optimal solution for MS problem is very expensive. And the algorithm is “superior”. In 2000, (Luo et al, 2000) offered a new approach for MS problems using fuzzy constraints, the authors mentioned that most of existing work in MS problems ignores the issue of fusing agents’ individual evaluations for a feasible time slot. They address this issue and suggest an axiomatic framework for the fusion operation. This work proposes a kind of selfish protocol for organizing negotiation among agents. The basic idea of the protocol is in some round of negotiation the coordinator agent propose a proposal. Other agents check the proposal with their own timetables, if the proposal cannot be accepted, the coordinator propose another one, the procedure continues until a proposal accepted by all agents or the coordinator cannot propose any more proposals. The authors claim that the proposed approach is “novel” compared to the previous work, since MS problem is modeled by FCSP in multi-agent environment, a kind of selfish protocol is presented and an axiomatic framework is identified for fusing agents’ preferences. In future work, the authors will develop other protocols for more complicated meeting scheduling problems and other models for fusing agents’ individual evaluations for a proposal.

4.1.1 Summary

Authors	Year	Name of the Paper	Approach	Main Contribution
Freuder and Wallace	1992	*Partial constraint satisfaction*	Standard constraint satisfaction problem	CSP solution techniques have analogues for solving partial constraint problems (PCSPs), which both cope with and take advantage of the differences between CSP and PCSP
Lemaitre and Verfaillie	1997	*An incomplete method for solving distributed valued constraint satisfaction problems*	Based on DVCSP to formalize MS problems	Incomplete method to solve DVCSP based on greedy repair centralized algorithm.
Tsuruta and Shintani	2000	*Scheduling meetings using distributed valued constraints satisfaction algorithm*	Distributed synchronous algorithm based on DVCSP	Discover a semi-optimal solution to over constrained MS problem,
Luo et al,	2000	*A multi-agent framework for meeting scheduling using fuzzy constraints*	Fuzzy Constraints	Selfish protocol is presented and axiomatic framework is identified for fusing agents'

Table 4.1.1 summarizing PCSP approaches.

4.2 User Preferences

Multi-agent meeting scheduling has been the means of studying relations among privacy, privacy loss, efficiency, and solution quality in multi-agent systems with preferences. Most studies of the meeting scheduling problem have included preference representations in their analysis and their systems (Sen and Durfe, 1995; Garrido and Sycara, 1996; Sen et al., 1997; Luo et al., 2000; Crawford and Veloso 2004; Franzin et al., 2004). In this section we review and discuss research efforts for solving MS problems based on user preferences and section 4.2.1 using privacy loss. The most interesting comparisons to be made regarding preferences pertain to the way they are combined to produce global evaluations. In most existing work on meeting scheduling, it is assumed that preference values for different agents can be combined directly (Ephrati et al., 1994; Garrido and Sycara, 1996; Luo et al., 2000).

(Ephrati et al., 1994) focus on two basic research problems in meeting scheduling. First, is the problem of timing - when to set a meeting. Second, how to choose the most appropriate time for particular meeting. The authors attempt to solve these problems by introducing three scheduling mechanisms for setting up meetings in closed system. Scheduling mechanisms are: Calendar Oriented Scheduling, Meeting Oriented approach and Schedule-Oriented Scheduling. All three mechanisms make use of primitive economics markets, where users assign “Convenience Points” to indicate their preferences over alternatives. Then each alternative examined to establish the group decision that maximizes utility. The authors claim that the more complex the mechanism is, the more it maintains the privacy of the users.

Similarly to (Garrido and Sycara, 1996), Ephrati et al. state that non of the earliest work is a truly autonomous agent view meeting scheduling as distributed task where each agent knows its user preferences and calendar availability in order to act on behalf of its user. The idea is to build a distributed system based on the (Sycara and Liu, 1994) system in which agents can exchange their meeting preferences and calendar information according to some privacy policy, each agent is able to relax three constraints: date, start-time and duration. In addition, each agent has weights that indicate how to relax each time constraint.

(Crawford and Veloso, 2004) argued that Ephrati et al 's three mechanisms approach are not considered as incentive compatible IC, due to the fact that their proof did not account for the repeated application of the clarke tax mechanism, rather it looked at single steps. As an alternative, Crawford and Veloso propose a mechanism in which each agent specifies 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. A schedule may be picked where some agents are not available for all meetings. The authors claim that their experimental results based on counter examples shows that the approaches of (Ephrati et al., 1994) are not incentive compatible IC. In future work, the authors would like to further explore the problem of IC in multi-agent meeting scheduling and exploring software agents that can learn peoples' scheduling preferences.

4.2.1 Privacy Loss Issues Based on User Preference

In recent years the issue of privacy has been considered in the field of distributed constraint satisfaction problem (Franzin et al., 2004). This topic has been discussed within the larger field of distributed artificial intelligence and in literature on meeting scheduling. Because privacy loss has not been a major concern, many systems use distributed protocols based on a single coordinator agent that collects all the useful information from other distributed agents (Scott et al., 1998).

Starting with (Freuder et al., 2001), the authors focus on important issue that arises in cooperative communication involving independent agents, which is privacy; there will be a cases where individuals will be interested in restricting the information communicated to other individuals. To measure efficiency of problem solving and privacy, the authors have implemented a multi-agent meeting scheduling system in which each agent has its own calendar, which consists of appointments in various cities at different days. The authors claim that when privacy concerns are overriding, no explicit information should be exchanged, but if efficiency is concern, the best method is to combine minimum of explicit information exchange with constraint-based inferences.

In 2004, (Franzin et al., 2004) built a meeting scheduling system based on an earlier (Freuder et al., 2001) system. The authors add preferences to the new system and observed the behavior of the new system under several conditions. Assuming that all agents try to maximize their preference subject to some global optimization criterion considering two basic global criteria: fuzzy optimality, having preference between 0 and 1, and maximizing the minimum preference. And pareto optimality, where a solution is optimal if there is no way to improve the preference of any agent without decreasing the preference of some other agent. From the observed result, the authors claim that fuzzy criterion can be used to lessen privacy loss with regard to preferences. And the pareto procedure can be used to minimize information exchange and privacy loss. This work is an extension to (Garrido and Sycara, 1996), by developing a method whereby agents can find a common meeting based on joint function of individual preferences without actually revealing them either before or after san agreement has been reached.

Recently, (Maheswaran et al., 2006) note that a general quantitative framework to compare existing metrics for privacy loss, and to identify dimensions along which to construct/classify new metrics, is currently lacking. The authors in this work develop a method to address these shortcomings In particular; this paper provides new additional experiments and analysis, detailed and formal descriptions of inference rules when detecting privacy loss. They refer to their previous work (Maheswaran, 2005) based on Valuation of Possible States (VPS) framework, which was designed as a quantitative model for comparing privacy loss metrics and developing new metrics. The authors present a VPS (Valuations of Possible States) framework, a general quantitative model from which one can analyze and generate metrics of privacy loss. VPS is shown to capture various existing measures of privacy created for specific domains of DisCSPs. The utility of VPS is further illustrated via analysis of privacy loss in DCOP algorithms, when such algorithms are used by personal assistant agents to schedule meetings among users. In addition, the authors develop techniques to analyze and compare privacy loss in DCOP algorithms; in particular, when using approaches ranging from decentralization (SynchBB (Hirayama and Yokoo, 1997), partial centralization (OptAPO (Mailler and Lesser, 2004)), as well as centralization. This involves constructing principled sets of inference procedures under various assumptions of knowledge by the agents. From their experimental evaluations the authors claim that decentralization by itself does not provide superior protection of privacy in DisCSP/DCOP algorithms, when compared with centralization. Instead, privacy protection requires the additional presence of uncertainty in agents' knowledge of the constraint graph. In future work, the authors state that they will continue to investigate algorithms or preprocessing strategies that improve DCOP solution efficiency if privacy is a major motivation for DCOP.

4.2.2 User Preference Based on Voting Scheme

Many research efforts have addressed user preference mechanism based on voting scheme and preference estimation (Sen and Durfee, 1991; Sen and Durfee, 1994; Sen and Durfee, 1996; Sen et al., 1997 and Chun et al., 2003).

Starting with (Sen and Durfee, 1991; Sen and Durfee, 1994) that was focusing on the problem of how an application domain for intelligent surrogate agents can be analyzed. One drawback of this work was they did not address many implementation issues like communication medium, user interaction, and use of preferences. In 1997, (Sen et al., 1997) realized that user preference is important and they modeled preferences as elections between different alternative proposals. To avoid conflicting of user preferences they used technique from voting theory which allows an agent to arrive at consensus choice for meeting times while balancing different user preferences. They have implemented distributed meeting scheduling system in a work-station-based computing environment, in this system the user interacts with the meeting scheduling system through the user interface. The authors claim that their autonomous scheduling can approximate the privacy and security concerns of users, and allows for better throughput and better fault tolerance.

In 2003, (Chun et al, 2003) stated that traditional optimal algorithms do not work without complete information about individual preferences, their work presents a new technique called “preference estimation” using “preference rules” that allow to find optimal solution to negotiations problems without needing to know the exact preference models of all the meeting participants beforehand, the authors describe and use two algorithms based on negotiation protocol. Algorithm 1 (NWOPI) and Algorithm 2 (NWPI) based on environment called Mobile Agent for Office Automation MAFOA (Wong et al., 2000). Simulations were performed to compare these algorithms. From the obtained results, the authors claim that algorithm 2, using preference estimation, finds the optimal solution at only a slightly higher cost that algorithm1, which relies on relaxation.

4.2.3 Summary

Authors	Year	Name of the Paper	Approach	Main Contribution
Ephrati et al.	1994	*A Non-manipulable Meeting Scheduling System*	Primitive economics markets	Additional user privacy could be maintained at the cost of decreased stability of the system
Garrido and Sycara	1996	*Multi-agent meeting scheduling: Preliminary experimental results*	Modeling and communication of constraints and preferences	MS performance is more stable and constant when agents try to keep their calendar and preference information private.

Sen et al	1997	*Satisfying User Preferences While Negotiating Meetings*	User preference based on Voting scheme	The proposed system can approximate the privacy and security concerns of users, and allows for better throughput and better fault tolerance
Freuder et al	2001	*Privacy/efficiency tradeoffs in distributed meeting scheduling by constraint-based agents*	Constraint-based inferences	When private information exchanged efficiency not improved unless sing o inference
Chun et al	2003	*Optimizing agent-based meeting scheduling through preference estimation*	Preference estimation based on the preference rules	Find optimal solution to negotiations problems without needing to know the exact preference models of all the meeting participants beforehand
Crawford and Veloso	2004	*Mechanism design for multi-agent meeting scheduling, including time preferences, availability, and value of prese*nce	Maximizing agents' utilities	Show how IR problem can be reduced, making mechanism design work in real-world is a theoretically challenging problem
Franzin et al.	2004	*Multi-Agent Constraint Systems with Preferences: Efficiency, Solution Quality, and Privacy Loss*	Fuzzy optimality, Pareto optimality	Minimize privacy loss, maximize solution quality, to be fast
Maheswaran		*Privacy loss in distributed constraint*	Valuation of	Privacy protection requires the presence of

et al.	2006	*reasoning: A quantitative framework for analysis and its applications*	Possible States (VPS) framework	uncertainty about agents' knowledge of the constraint graph

Table 4.3 summarizing user preference methods.

4.3 Distributed Constraints Satisfaction Problem

Distributed Constraint Satisfaction Problems (DisCSP) have long been considered an important area of research for multi-agent systems (Mailler and Lesser, 2003). This is partly due to the fact that many real-world problems can be represented as a constraint satisfaction problem (Russell and Norvig, 2003). However there are not many published efforts on DisCSP to solve the meeting-scheduling problem. Only a few works related to DisCSP in a multi-agent system can be found in the literature (Luo et al., 1992; Yokoo and Hirayama, 1996, 1998; Mailler and Lesser, 2003).

In 1992, (Luo et al., 1992) presented a major decomposition technique based on breaking apart the search space by assigning particular domain elements from one or more of the variables to individual agents. One major drawback of this technique is that each of the agents has to know the variables, domains, and constraints for the entire problem. The other DisCSPs work (Yokoo and Hirayama, 1996) which present variable decomposition technique that involves assigning each agent one or more variables to manage giving each knowledge of the constraints on their variables.

In 1998, Yokoo and Hirayama stated that previous algorithms to solve distributed constraint satisfaction problems (DisCSPs) are neither efficient nor scalable to larger problems, since they assume each agent has only one local variable. In this work (Yokoo and Hirayama, 1998) the authors intend to develop a better algorithm that can handle multiple local variables efficiently based on Asynchronous Weak-Commitment search algorithm. They referred to their previous work (Yokoo, 1995) in which the priority order of agents is changed dynamically. One limitation of this work as it mentioned by the author is the algorithm assume each agent has only one local variable, and this assumption cannot be satisfied when the local problem of each agent becomes large and complex. The proposed algorithm based on Asynchronous Weak-Commitment search algorithm in which each agent sequentially performs the computation for each variable, and communicates with other agents only when it can find a local solution that satisfies all local constraints. By using this algorithm bad local solutions can be modified without forcing other agents to exhaustively search local problems, which lead to decrease the number of interactions among agents. From their experimental evaluations the authors claim that the proposed algorithm is far more efficient than their previous algorithm (Yokoo, 1995) that uses the prioritization among agents. Another attempt to overcome the previous limitations of (Yokoo, 1995) and solving DisCSPs was by (Mailler and Lesser, 2003) which present better method of cooperative mediation that allows the agents to extend and overlap the context that they use for making their local decisions. The proposed method is based on a negotiation protocol called Asynchronous Partial Overlay (APO) algorithm. The idea of this algorithm is that agents mediate over conflicts, the context they use to make local decisions overlaps with that of the other agents, the agents gain more context information along with critical paths of constrain graph to improve their decision. To evaluate the proposed algorithm the authors implement APO algorithm and compare the results with AWC algorithm (Yokoo and Hirayama, 2000). From the conducted experiments the authors claim that APO algorithm is both “sound and complete”, and performs better than AWC algorithm for both sparse and critical graph coloring problems.

In 2005, Ferreira and Bazzan state that a previous approach (Mailler and Lesser, 2003) yields good results in a simple scenarios, but there is a lack of analysis in complex real-world ones such as, distributed meeting scheduling problem. In their work (Ferreira and Bazzan, 2005) discuss the difficulties of applying the cooperative mediation OptAPO algorithm in real-world problem. The authors use Distributed meeting scheduling problem mapped as Distributed Constraint Optimization Problem

DCOP using DiMES (Maheswaran et al., 2004) to solve MS problem. And compare the performance of the OptAPO with adopt algorithms, then they proposed use of heuristic search mechanism to replace branch-and-bound search (B&B) in the cooperative mediation. They claim that the obtained results are "very promising" the heuristic version of OptAPO achieves the best solution with significant better performance, outperforming Adopt even with speedup heuristics. In future work, the authors intend to pursue three directions. First, run the proposed approach with different scenarios. Second, compare the results of proposed approach with other incomplete, heuristics DCOP algorithms. Finally, investigate which classes of problems are adequate for which type of DCOP algorithm.

4.3.1 Summary

Authors	**Year**	**Name of the Paper**	**Approach**	**Main Contribution**
Luo et al	1992	*A hybrid algorithm for distributed constrains satisfaction problems*	DisCSP	Assigning particular domain elements from one or more of the variables to individual agents
Yokoo and Hirayama	1996	*Distributed breakout algorithm for solving distributed constraint satisfaction problems*	Modeling and communication of constraints and preferences	Assigning each agent one or more variables to manage giving each knowledge of the constraints on their variables
Yokoo and Hirayama	1998	*Distributed Constraint Satisfaction Algorithm for Complex Local*	Asynchronous Weak-Commitment search	Bad local solution can be modified without forcing other agents to exhaustively search local problems, decrease the

		Problems		number of interactions among agents.
Mailler and Lesser	2003	*A mediation based protocol for distributed constraint satisfaction*	Cooperative mediation	Solving DisCSPs by developing cooperative mediation protocol based on negotiation protocol APO
Ferreira and Bazzan	2005	*Distributed meeting schedule through cooperative mediation*	DCOP based on DiMES	The heuristic version of OptAPO achieves the best solution with significant better performance, outperforming Adopt even with speedup heuristics.

Table 4.3.1 Summarizing DCSP approaches.

4.4 Constraint Logic Programming

Constraint Logic Programming has been a promising approach for solving scheduling problems (Fruhwirth and Abdennadher, 1997; Marriot and Stuckey, 1998; and Abdennadher and Schlenker, 1999). CLP combines the advantages of two declarative paradigms: logic programming and constraint solving, in logic programming, problems are tested in declarative way using rules to define relations (predicates). Problems are solved using chronological backtrack search to explore choices. In constraint solving, efficient special-purpose algorithm are employed to solve sub-problems involving distinguished relations referred to as constraints, which can be considered as pieces of partial information.

(Abdennadher and Schlenker, 1999) state that there is no general method exists for solving efficiently many real life problems that lead to combinatorial search, such as automatic generation of duty roster for hospital wards, the authors attempt to solve this problem using Constraints Logic Programming (CLP) framework, to model nurse scheduling problem as partial constraint satisfaction problem in CLP framework, the authors referred to (Freuder and Wallace, 1992), this works dealing with soft constraints by proposing Hierarchical Constraint Logic Program (HCLP) approach that support hierarchical organization of constraints. While another work by (Meyer, 1997) avoids the inter-hierarchy comparison in HCLP, the soft constraints are encoded in Hierarchical Constraints Satisfaction Problem (HCSP). Nurse-scheduling problem can be modeled as a partial constraint satisfaction problem that requires processing of hard and soft constraints to cope with. Hard constraints are conditions that must be satisfied, soft constraint maybe violated, but should be satisfied as far as possible. To evaluate CLP approach, the authors have developed INTERDIP system that implemented with Siemense-Nixdorf-Informationssysteme using IF/prolog that includes a constraints package. INTERDIP has been successfully tested on a real ward “Klinikum Innenstadt” hospital in Munich, Germany. The authors claim that the generated schedules using INTERDIP are “better” comparable to those manually generated by a well-experienced head nurse.

4.5 The DRAC model

In 2004, BenHassine argued that most of existing approaches to solve MS problems are centralized CSP such as (Abdelnnadher and Schlenker, 1999; Bakker et al., 1993). And he claimed that MS problem is naturally distributed and it cannot be solved by centralized approach. Other researchers (Garrido and Sycara, 1996) focused on using distributed autonomous and independent agent to solve MS problem where each agent knows its user preferences and calendar availability. However, (BenHassine et al., 2004) mentioned that the majority of existing works on MS tackle it as static problem, allow for relaxation of any constraints and do not deal with achieving any level of consistency. In attempt to overcome these limitations the authors present new distributed approach MSRAC based on DRAC model (Distributed Reinforcement of arc Consistency). The basic idea of this approach consists of two steps. The first reduces the initial problem by reinforcing some level of local consistency. The second step solves the resulting MS problem while maintaining arc-consistency. The authors have developed the multi-agent dynamic with Acttalk, using the Smalltalk-80 environment with generating random meeting problems, then they compared their approach with other approaches including the Asynchronous Backtracking approach ABT (Yokoo and Hirayama, 2000) and Tsuruta'a approach (Tsuruta and Shintani, 2000). They claim that the obtained results show the MSRAC approach requires in the majority of cases less CPU time than other approaches. As for the numbered of scheduled meetings, ABT and MSRAC schedule almost the same number of meetings. In future work the authors will try to integrate learning process, and implementing the improved approach on a cluster of computer.

5 CONCLUDING COMMENTS

This survey is designed to serve as an introduction to people who are interested in the meeting-scheduling problem that happened dynamically based on multi-agent systems. In this survey we have described the research efforts on several approaches to solve world-distributed meeting-scheduling problem. However, meeting scheduling is a time-consuming, repetitive, and essential part of daily chores in an organization. It involves searching for a time and place when and where all the meeting participants are free and available.

In this survey, many researchers state that meeting scheduling problem cannot be solved by centralized approach due to its dynamic features, it needs to cooperate with distributed agents to reach an agreement of coordination in order to achieve scheduling task with minimal cost. Other researchers have used user preference in evaluating scheduling proposals in distributed agent-based meeting scheduling by using preference estimation techniques that allows optimal meeting schedules to be found without complete knowledge of individual user preference models. Others have developed a formal model for distributed meeting scheduling that can be used to make predictions about the impact of various heuristic strategies.

Another novel aspect of present work is the introduction of articulated agent views that can be brought to bear on the solving process, this has no clear antecedent in previous work on meeting scheduling. This feature can be used to enable agents to organize and update information about other agents in coherent and different manner.

APPENDIX-I

Note: The paper with symbol "*" is a milestone paper; the paper with symbol "^" is an important paper.

Bibliography

[1].^(Abdennadher and Schlenker, 1999) Abdennadher S and Schlenker H (1999) Nurse scheduling using constraint logic programming. In: *Proceedings of (IAAI)*, 838–843.

[2]. (Bakker et al. , 1993) Bakker R, Dikker F, Tempelman F, and Wongum P (1993) Diagnosing and solving over-determined constraint satisfaction problems, In: *Proceedings of 13th International Joint Conference on Artificial Intelligence (IJCAI),* 276-281.

[3].^(BenHassine et al. , 2006) BenHassine A, Ho T, and Ito T (2006) Meeting scheduling solver enhancement with local consistency reinforcement, In: *the International Journal of Applied Intelligence, 24(2) Springer*, 143-154.

[4].*(BenHassine et al. , 2004) BenHassine A, Ho T, and Ito T (2004) Agent-Based Approach to Dynamic Meeting Scheduling Problems. In: *Proceedings of 3rd International ACM Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS)***,** 1132-1139.

[5]. (BenHassine, 2005) BenHassine A (2005) Distributed Reinforcement of local Consistency for General Constraints Network An Investigation of Meeting Scheduling Problems, P.hD thesis, Japan Advanced Institute of Science and Technology.

[6].^(Chun et al. , 2003) Chun A, Wai H, and Wong R (2003) Optimizing agent-based meeting scheduling through preference estimation. In: *Engineering Applications of Artificial Intelligence, 16(7)*, 727-743.

[7].^(Crawford and Veloso, 2004) Crawford E and Veloso M (2004) Mechanism design for multi-agent meeting scheduling, including time preferences, availability, and value of presence. In: *Proceedings of International Conference on Intelligent Agent Technology,* 253- 259.

[8]. (Davis and Patterson, 1975) Davis E and Patterson J (1975) A comparison of heuristic and optimum solution in resource-constrained project scheduling, *Management Science 21*, 994-955.

[9]. (Dumond and Mabert, 1988) Dumond J and Mabert V (1988) Evaluating projects scheduling and due date assignment procedures: an experimental analysis, *Management Science 34*, 101-118.

[10]. (Eaton et al., 1998) Eaton P, Freuder E, and Wallace R (1998) Constraints and Agents Confronting ignorance. *AI magazine 19(2)*, 51-65.

[11].^(Ephrati et al. , 1994) Ephrati E, Zlotkin G, and Rosenschein J (1994) A Non-manipulable Meeting Scheduling System, In: *proceedings of 13th International Distributed Artificial Intelligence Workshop*, 105-125.

[12].*(Ferreira and Bazzan, 2005) Ferreira J and Bazzan A (2005) Distributed meeting schedule through cooperative mediation. In: *Proceedings of 6th International Workshop on Distributed Constraints Reasoning*.

[13].^(Franzin et al. , 2004) Franzin M, Rossi F, Freuder E, and Wallace R (2004) Multi-Agent Constraint Systems with Preferences: Efficiency, Solution Quality, and Privacy Loss, In: *Computational Intelligence 20 (2)*, 264-286.

[14].*(Freuder et al. , 2001) Freuder E, Minca M, and Wallace R (2001) Privacy/efficiency tradeoffs in distributed meeting scheduling by constraint-based agents. In: *Proceedings of (IJCAI) Workshop on Distributed Constraint Reasoning*, 63-71.

[15]. (Freuder and Wallace, 1992) Freuder E and Wallace R (1992) Partial constraint satisfaction, In: *Proceedings of the 11th International Joint Conference on Artificial Intelligence. 58 (1-3)*, 21-70.

[16].(Fruhwirth and Abdennadher,1997) Fruhwirth T and Abdennadher S (1997) Constraint Programmierung. (in German),Textbook, Springer-Verlag.

[17]. (Hirayama and Yokoo, 1997) Hirayama K and Yokoo M (1997) Distributed partial constraint satisfaction problem. In: *G. Smolka (ed.):*

Principles and Practice of Constraint Programming. 222–236.

[18].*(Garrido and Sycara, 1996) Garrido L and Sycara K (1996) Multi-agent meeting scheduling: Preliminary experimental results. In: *Proceedings of 1st International Conference on Multi-Agent Systems (ICMAS)*. 95 – 102.

[19]. (Garrido et al., 1996) Garrido L, Brena R, and Sycara K (1996) Cognitive Modeling and Group Adaptation in Intelligent Multi-Agent Meeting Scheduling. In: *Proceedings of 1st Iberoamerican Workshop on Distributed Artificial Intelligence and Multi-Agent Systems*, 55 – 72.

[20]. (Jennings, 2000) Jennings N (2000) On agent-based software engineering. *In: Artificial intelligence, 117(2),* 277-296.

[21]. (Kelly and Chapanis, 1987) Kelly J and Chapanis A (1987) How professional persons keep their calendars: Implication for computerization. In: *Journal of Occupational Psychology 55*, 141-156.

[22]. (Lemaitre and Verfaillie, 1997) Lemaitre M and Verfaillie G (1997) An incomplete method for solving distributed valued constraint satisfaction problems. In: *Proceedings of AAAI workshop on constraints and agents.*

[23].*(Luo et al. , 2000) Luo X, Leung H, and Lee J (2000) A multi-agent framework for meeting scheduling using fuzzy constraints. In: *Proceedings of the 4th International Conference on MultiAgent Systems*, 409–410.

[24]. (Luo et al., 1992) Luo Q, Hendry P, and Buchanan J (1992) A hybrid algorithm for distributed constrains satisfaction problems. In: *Proceedings of the 9th European workshop on Parallel Computing (EWPC).*

[25].^(Maheswaran et al., 2006) Maheswaran R, Pearce J, Bowring E, Varakantham P, and Tambe M (2006) Privacy loss in distributed constraint reasoning: A quantitative framework for analysis and its applications. In: *Autonomous Agents and Multi-Agent Systems (JAAMAS)*, 27 – 60.

[26]. (Maheswaran et al., 2005) Maheswaran R, Pearce J, Varakantham P, Bowring E, and Tambe M (2005) Valuations of Possible States (VPS): A unifying quantitative framework for analysis of privacy

loss in collaboration. In: *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS05)*, 1030–1037.

[27]. (Maheswaran et al., 2004) Maheswaran R, Tambe M, Bowring E, Pearce J, and Varakantham P (2004) Taking DCOP to the real world: Efficient complete solution for distributed multi-event scheduling. In: *Proceedings of 3rd international Joint Conference on Autonomous Agents and Multiagent Systems, Volume 1*, 310-317.

[28]. (Mailler and Lesser, 2004) Mailler R and Lesser V (2004) Solving distributed constraint optimization problems using cooperative mediation. In: *Proceedings of 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 04)*, 438–445.

[29].^(Mailler and Lesser, 2003) Mailler R and Lesser V (2003) A mediation based protocol for distributed constraint satisfaction. In: *Proceedings of the 4th International Workshop on Distributed Constraint Reasoning*, 49-58.

[30]. (Marriot and Stuckey, 1998) Marriot K and Stuckey P (1998) Programming with Constraints: An Introduction. The MIT Press.

[31]. (Meyer**,** 1997**)** Meyer H (1997) ConPlan/SIED Aplan: Personnnel assignment as a problem of hierarchical constraint satisfaction. In: *Proceedings of the 3rd International Conference on the Practical Application of Constraint Technology.*

[32].^(Modi et al. , 2004) Modi P, Veloso M, Smith S, Cmradar J, and Oh J (2004) A personal assistant agent for calendar management. In: *Proceedings of 6th International Workshop on Agent-Oriented Information Systems (AOIS)*, 134-148.

[33].*(Modi and Veloso, 2004) Modi P and Veloso M (2004) Multiagent Meeting Scheduling with Rescheduling. In: *Proceedings of 5th Workshop on Distributed Constraint Reasoning (DCR).*

[34]. (Noronha and Sarma, 1991) Noronha S and Sarma V (1991) Knowledge-based approaches for scheduling problems: A survey. *In IEEE Transactions on Knowledge and Data Engineering, 3(2)*, 160-171.

[35]. (Russell and Norvig, 2003) Russell S and Norvig P (2003) Artificial Intelligence A Modern Approach. Second edition, chapter 2, Prentice Hall.

[36]. (Scott et al., 1998) Scott A, Jenkin J, and Senjen R (1998) Design of An-Agent based, Multi-user Scheduling Implementation. In: *Multi-agent system-theories Languages and Applications, Lecture Notes in Artificial Intelligence, 1544*, 152-165.

[37].^(Sen et al., 1997) Sen S, Haynes T, and Arora N (1997) Satisfying User Preferences While Negotiating Meetings. In: *International Journal of Human-Computer Studies 47*, 407-415.

[38].^(Sen and Durfee, 1996) Sen S and Durfee E (1996) A Contracting Model for Flexible Distributed Scheduling. In: *Annals of Operations Research, 65*, 195–222.

[39]. (Sen and Durfee, 1995) Sen S and Durfee E (1995) Unsupervised surrogate agents and search bias change in flexible distributed scheduling. In: *Proceedings of the first international conference on Multi-Agents System, ICMAS'95*, 336-343.

[40]. (Sen and Durfee, 1994) Sen S and Durfee E (1994) On the design of an adaptive meeting scheduler. In: *Proceedings of 10th IEEE Conference on Artificial Intelligence Application*, 40-46.

[41].*(Sen and Durfee, 1991) Sen S and Durfee E (1991) A formal study of distributed meeting scheduling : Preliminary Results. In: *Proceeding of the ACM conference on Organization Computing System*, 58-68.

[42]. (Sycara and Liu, 1994) Sycara K and Liu J (1994) Distributed meeting scheduling. In: *16th Annual Conference of the Cognitive Society,* 449-461.

[43].^(Tsuruta and Shintani, 2000) Tsuruta T and Shintani T (2000) Scheduling meetings using distributed valued constraint satisfaction algorithm. In: *Proceedings of the 14th European Conference on Artificial Intelligence-(ECAI 2000)*, 383-387.

[44]. (Wong et al., 2000) Wong Y, Ho T, Fung K, and Chun H (2000) A model for resources negotiation using mobile agents. In: *Proceedings of 4th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2000)*,

[45]. (Wooldridge, 2002) Wooldridge M. (2002) An Introduction to MultiAgent Systems. John Wiley.

[46].(Wooldridge and Jennings,1995) Wooldridge M, Jennings N (1995) intelligent agent: theory and practice. In: *The knowledge engineering review, 10(2)*, 115-152.

[47]. (Yokoo and Hirayama, 2000) Yokoo M and Hirayama K (2000) Algorithms for distributed Constraints Satisfaction : A Review. In: *Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS) 3(2)*, 198-212.

[48].^(Yokoo and Hirayama, 1998) Yokoo M and Hirayama K (1998) Distributed Constraint Satisfaction Algorithm for Complex Local Problems. In: *Proceedings of 3rd International Conference on Multiagent Systems (ICMAS-98)*, 372–379.

[49]. (Yokoo and Hirayama, 1996) Yokoo M and Hirayama K (1996) Distributed breakout algorithm for solving distributed constraint satisfaction problems. In: Proceedings *of 1st International Conference on Multi-Agent Systems (ICMAS)*, 401- 408.

[50]. (Yokoo, 1995) Yokoo M (1995) Asynchronous weak-commitment search for solving distributed constraint satisfaction problems. In: *Proceedings of the 1st International Conference on Principles and Practice of Constraint Programming (CP-95)*, 88–102.

[51]. (Radar Project, 2004) The Radar Project. www.radar.cs.cmu.edu, 2004.

APPENDIX-II

Annotations to the bibliography

Note: The paper with symbol "*" is a milestone paper; the paper with symbol "^" is an important paper.

[1]. ^(Abdennadher S, Schlenker H, 1999)

The authors state that there is no general method exists for solving efficiently many real life problems that lead to combinatorial search, such as automatic generation of duty roster for hospital wards, in this paper the authors attempt to solve this problem using Constraints Logic Programming (CLP) framework. to model nurse scheduling problem as partial constraint satisfaction problem in CLP framework, the authors referred to (Freuder E and Wallace R, 1992), this works dealing with soft constraints by proposing Hierarchical Constraint Logic Program (HCLP) approach that support hierarchical organization of constraints. While another work by (Meyer H, 1997) avoids the inter-hierarchy comparison in HCLP, the soft constraints are encoded in Hierarchical Constraints Satisfaction Problem (HCSP). Nurse-scheduling problem can be modeled as a partial constraint satisfaction problem that requires processing of hard and soft constraints to cope with. Hard constraints are conditions that must be satisfied, soft constraint maybe violated, but should be satisfied as far as possible. to evaluate CLP approach, the authors have developed INTERDIP system that implemented with Siemense-Nixdorf-Informationssysteme using IF/prolog that includes a constraints package. INTERDIP has been successfully tested on a real ward "Klinikum Innenstadt" hospital in Munich, Germany. The authors claim that the generated schedules using INTERDIP are "better" comparable to those manually generated by a well-experienced head nurse.

Citation:

BenHassine A, Ho TB, Ito T (2004) Agent-Based Approach to Dynamic Meeting Scheduling Problems. In: Proc AAMAS

[2] ^(BenHassine A et al. , 2006)

The problem addressed by the authors was to develop a better method for scheduling all the meetings while satisfying all the constraints related to both users and meetings in order to reach an agreement between all participants. Recent researchers have argued that the best way to solve MS problem is to use an agent-based approach, one of these researchers is (Ephrati E et al., 1994) this work defined two paradigms of MS scenarios, open scheduling systems and closed scheduling systems. While (Garrido L and Sycara K, 1995) reported another MAS work that focus on using distributed autonomous and independent agents to solve the problem.

The authors proposed to formalize the MS problem as (VCSP) in which each user maintains two kinds of constraints: hard and soft constraints related to themselves, in addition to the other constraints defining the problem. The hard constraints represent non-availability of the user, while the soft constraints represent the preference calendar of a user. This approach was implemented with Actalk under the Smalltalk-80 environment and compared with an existing approaches on randomly generated instances in mean of CPU time, percentage of scheduled meetings and the number of exchange messages. The obtained results show that this approach is scalable and worthwhile for processing strong constraints. The authors claim that this approach is appropriate for reducing the MS problem and consequently the search space, without loss of solution. Improving this approach by allowing a more relaxed system, and implementing the improved approach on multi-processor platform using real data is what the authors trying to do in their future work.

[3] *(BenHassine A et al. , 2004)

The problem addressed by authors that the majority of existing works on MS tackle it as static problem, allow for relaxation of any constraints and do not deal with achieving any level of consistency. In attempt to overcome these limitations the authors in this paper present new distributed approach MSRAC based on DRAC model (distributed Reinforcement of arc Consistency). Recent research adopts the agent based approach for solving MS problem among these works, first work proposed by (Sen S et al., 1997), this work concentrate on problem of how an application domain for intelligent surrogate agents can be analyzed, understood and represented, the authors have considered MS problem as domain for their main research, their efforts were directed towards the integration of user preference. While the second work presented by (Tsuruta T and ShintaniT , 2000) uses a group of schedule management system based on multiple agents. The authors in this paper proposed Distributed Valued Constraints Satisfaction Problem (DVCSP) formalism to model MS problem. The basic idea of this approach as authors presented in this paper consists of two steps. The first reduces the initial problem by reinforcing some level of local consistency. The second step solves the resulting MS problem while maintaining arc-consistency. The authors have developed the multi-agent dynamic with Acttalk, using Smalltalk-80 environment with generating random meeting problems. They have experimented two approaches, Asynchronous Backtracking (Yokoo M and Hirayama K, 2000) and Tsuruta'a approach (Tsuruta T and Shintani T, 2000), the results in this work are expressed in terms of three criterions: CPU time (in millisecond), the number of scheduled meetings and number of exchanged messages. The authors claim that their MSRAC approach is scalable and performs less message passing for almost the same solutions. Integrating learning process, and trying to implement the improved approach on a cluster of computer, these are what the authors trying to achieve in their future work.

Citation: this paper cited by:

Maheswaran R, Pearce J, Bowring E, Varakantham E (2006) *Privacy Loss in Distributed Constraint Reasoning: A Quantitative Framework for Analysis and its Applications* , Autonomous Agents and Multi-Agent Systems, Springer.

[4] ^(Chun A et al. , 2003)

The authors state that a current traditional optimal algorithm does not work without complete information about individual preferences, this paper presents new technique that allow to find optimal solution to negotiations problems without needing to know the exact preference models of all the meeting participants beforehand. They referred to (Wong et al., 2000) this work present an environment called Mobile Agent for Office Automation MAFOA, the agent based algorithm that proposed in this paper work within this environment. To design an optimizing algorithm that can perform distributed scheduling without complete knowledge of individual models, the authors used technique called “preference estimation” used the “preference rules” that allow preferences to change dynamical as scheduling decisions are made. This mimics changing preferences as schedule gets filled. They described and used two algorithms based on negotiation protocol, Algorithm 1 (NWOPI) and Algorithm 2 (NWPI). Simulations performed to compare these algorithms, from the obtained result the authors claim that algorithm 2 using preference estimation, finds the optimal solution at only a slightly higher cost that algorithm1, which relies on relaxation.

Citation:

Lee W (2004) *Towards agent-based decision making in the electronic marketplace: interactive recommendation and automated negotiation.* Expert Systems With Applications – Elsevier.

[5] ^(Crawford E, Veloso M, 2004)

The authors in this paper state that scheduling multiple meeting with many participant possibly in parallel is cognitively demanding, the authors design better mechanism that enables such agent to effectively schedule meetings and assign the meetings to times in a way that maximizes social welfare. They referred to existing mechanisms design approach to multi-agent meeting scheduling (Ephrati et al., 1994) this work used a Clark tax mechanism to try and ensure IC. Three different approaches were taken to implement the mechanism: meeting-oriented; calendar-oriented and schedule-oriented.

In this paper the authors propose mechanism that each agent specifies 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. A schedule may be picked where some agents are not available for all meetings. the authors claim that their results demonstrate that making mechanism design work in real-world multi-agent systems is a theoretically challenging problem. In the future work, the authors would like to further explore the problem of IC in multi-agent meeting scheduling and exploring software agents that can learn people’s scheduling preferences.

Citation: this paper cited by:

Baroni P, Gerevini A, Toninelli P, (2005) *MAgentA: Un Sistema Multi-Agente per la Gestione di Agende e Riunioni.* In: Proceedings of the WOA05.

[6] ^(Ephrati E et al. , 1994)

This paper focus on two basic research problems in meeting scheduling. First, problem of timing, when to set a meeting. Second, how to choose the most appropriate time for particular meeting. The authors attempt to solve these problems by introducing three scheduling mechanisms for setting up meetings in closed system. The authors referred to (Noronha S and Sarma V, 1991) this work focused on problem of timing, when to set a meeting, specially if there are several meetings to be scheduled that involves the same resources, and as the number of participants grows.

The authors have introduced in this paper three scheduling mechanisms: Calendar Oriented Scheduling, Meeting Oriented approach and Schedule Oriented Scheduling. All three mechanisms make use of primitive economics markets, where users assign "Convenience Points" to indicate their preferences over alternatives. Then each alternative examined to establish the group decision that maximizes utility. The authors claim that the more complex the mechanism is, the more it maintains the privacy of the users. Also they have shown from the obtained results that all proposed mechanisms are approximate the optimal utilitarian choice, and find a solution that is pareto optimal and is the condorcet winner.

Citation:

Crawford E, Veloso M (2004) *Mechanism design for multi-agent meeting scheduling, including time preferences, availability, and value of presence*. In: Proc.IEEE/WIC/ACM Int.Con. on Intelligent Agent Technology.

[7] *(Ferreira J, Bazzan A, 2005)

The authors' state that most of previous approaches to solve DCOP yield good results in simple scenarios, but there is a lack of analysis in complex real-world ones such as, distributed meeting scheduling problem. In this paper the authors discuss the difficulties of applying the cooperative mediation OptAPO algorithm in real-world problem. They referred to an existing approach based on cooperative mediation proposed by (Mailler R and Lesser V, 2004), this approach propose new way to find optimal solution to DCOP and it is the basis of the OptAPO algorithm. Furthermore the authors show that OptAPO performs better than Adopt. The authors in this paper use distributed meeting scheduling problem mapped as DCOP using DiMES (Maheswaran R et al., 2004) to solve MS problem. And compare the performance of the OptAPO and adopt algorithms, then they used heuristic search mechanism to replace branch-and-bound search (B&B) in the cooperative mediation. The authors' claim that the obtained results are "very promising": the heuristic version of OptAPO achieves the best solution with significant better performance, outperforming Adopt even with speedup heuristics. In future work, the authors intend to pursue three directions. First, to run the proposed approach with different scenarios. Second, to compare the results of proposed approach with other incomplete, heuristics DCOP algorithms. Finally, investigate which classes of problems are adequate for which type of DCOP algorithm.

[8] ^(Franzin M et al. , 2004)

The authors state that agents need to communicate and exchange information to obtain global solution. But these agents need to find a good global solution in a short amount of time, to handle these issues the authors in this paper develop formalization for multi-agent constraint system with preferences that is related to basic constraint solving situation. They referred to many previous studies that do not develop protocols for finding globally optimal solution (Garrido and Sycara K, 1996; Sen S and Durfee E, 1995), in this work negotiation process stops as soon as all agent have accepted a proposal because these protocols use satisficing criterion.

The authors' in this paper assume that all agents try to maximize their preference subject to some global optimization criterion considering two basic global criteria: fuzzy optimality, having preference between 0 and 1, and maximizing the minimum preference. And pareto optimality, where a solution is optimal if there is no way to improve the preference of any agent without decreasing the preference of some other agent. From their comparison experimental results the authors claim that fuzzy criterion can be used to lessen privacy loss with regard to preferences. And pareto procedure can be used to minimize information exchange and privacy loss.

[9] *(Freuder E et al. , 2001)

The authors in this paper focus on important issue that arises in cooperative communication involving independent agents, which is privacy; there will be a cases where individuals will be interested in restricting the information communicated to other individuals. This paper presents study of privacy/efficiency tradeoff in context of meeting scheduling. The authors referred to one area of application receiving considerable attention is Meeting Scheduling (Eato et al., 1998; Garrido L and Sycara K, 1996) this is a task which might be profitably delegated to software agents communicating over networks. To measure efficiency of problem solving and privacy, the authors have implemented a multi-agent meeting scheduling system in which each agent has its own calendar, which consists of appointments in various cities at different days. From the obtained result the authors shows that when more private information is exchanged, overall efficiency is not improved unless agents use some means of inference to reduce the search further. The authors claim that when privacy concerns are overriding, no explicit information should be exchanged, but if efficiency is concern, the best method is to combine minimum of explicit information exchange with constraint-based inferences.

Citation:

Yokoo M, Suzuki K, Hirayama K (2002) *Secure distributed constraint satisfaction: Reaching agreement without revealing private information*, Proc. of the AAMAS-02 DCR Workshop – Springer.

[10] *(Garrido L, Sycara K, 1996)

The authors state that non of the earliest work is a truly autonomous agent, in this paper the authors attempt to view meeting scheduling as distributed task where each agent knows its user preferences and calendar availability in order to act on behalf of its user.

They referred to several works in distributed meeting scheduling, such (Sycara K and Liu J, 1994) this work present an approach based on modeling and communication of constraints and preferences among the agents. Using this model, agents can react to dynamic changes. The idea as presented in this paper, based on the distributed system by (Sycara K and Liu J, 1994) agents can exchange their meeting preferences and calendar information according to some privacy policy, each agent is able to relax three constraints: date, start-time and duration. In addition, each agent has weights that indicate how to relax each time constraint, the authors have implemented a simplified version of the coordination mechanism and communication protocol from (Sycara K and Liu J, 1994) the experiments show some of the relationships between different experimental variables, such as calendar and preference privacy. From the obtained results the authors claims that meeting scheduling performance is more stable and constant when agents try o keep their calendar and preference information private. In future work, the authors intend to let agents learn and infer other agents' mental attitudes and behavior in order to model more complex and realistic scenarios.

Citation: this paper cited by:

Franzin M, Rossi F , Freuder E , Wallace R (2004) *Multi-Agent Constraint Systems with Preferences: Efficiency, Solution Quality, and Privacy Loss*, Computational Intelligence 20 (2), 264-286

[11]. *(Luo X et al. , 2000)

The authors mentioned that most of existing work in MS problems ignores the issue of fusing agents' individual evaluations for feasible time slot. While the authors in this paper address this issue and suggest an axiomatic framework for fusion operation. They referred to (Scott A et al., 1998) where a sort of fusion operator is involved and they are just some specific operator rather than an axiomatic framework. One drawback of this work they do not put weights of agents into account when fusing.

The authors in this paper propose a kind of selfish protocol for organizing negotiation among agents. The basic idea of the protocol is, in some round of negotiation the coordinator agent propose a proposal. Other agents check the proposal with their own timetables, if the proposal cannot be accepted, the coordinator propose another one. The procedure continues until a proposal accepted by all agents or the coordinator cannot propose any more proposals. The authors claim that the proposed approach is "novel" compared to the previous work, since MS problem is modeled by FCSP in multi-agent environment, a kind of selfish protocol is presented and axiomatic framework is identified for fusing agents'. In future work the authors attempt to develop other protocols for more complicated meeting scheduling problems and other model for fusing agents' individual evaluations for a proposal.

Citation:

Luo X, Jennings N, Shadbolt N, Leung H, Lee J (2002) *A fuzzy constraint based model for bilateral,multi-issue negotiations in semi-competitive Environments*. Artificial Intelligence, 148 pp 53–102 - Elsevier.

[12]. ^(Maheswaran R et al. , 2006)

The authors state that a general quantitative framework to compare existing metrics for privacy loss, and identify dimensions along which to construct/classify new metrics is currently lacking. The authors in this work develop a method to address these shortcomings In particular; this paper provides new additional experiments and analysis, detailed and formal description of inference rules when detecting privacy loss, they referred to their previous work (Maheswaran R, 2005) that based on Valuation of Possible States (VPS) framework, which was designed as a quantitative model for comparing privacy loss metrics and developing new metrics, the authors presents VPS (Valuations of Possible States) framework, a general quantitative model from which one can analyze and generate metrics of privacy loss. VPS is shown to capture various existing measures of privacy created for specific domains of DisCSPs. The utility of VPS is further illustrated via analysis of privacy loss in DCOP algorithms, when such algorithms are used by personal assistant agents to schedule meetings among users. Second, the authors develop techniques to analyze and compare privacy loss in DCOP algorithms; in particular, when using approaches ranging from decentralization (SynchBB (Hirayama and Yokoo, 1997), partial centralization (OptAPO (Mailler and Lesser, 2004)), as well as centralization. This involves constructing principled sets of inference procedures under various assumptions of knowledge by the agents. From their experimental evaluations the authors claim that decentralization by itself does not provide superior protection of privacy in DisCSP/DCOP algorithms when compared with centralization; instead, privacy protection requires the additional presence of uncertainty in agents' knowledge of the constraint graph. In future work the authors will continue to investigate algorithms or preprocessing strategies that improve DCOP solution efficiency. If privacy is a major motivation for DCOP.

Citation:

Greenstadt R, Pearce J, Bowring E, Tambe M (2006) *Experimental analysis of privacy loss in DCOP algorithms.* In: Proceedings of the 5th international joint conference on Autonomous agents and multiagent systems, pp. 1424 – 1426.

[13]. ^(Mailler R, Lesser V, 2003)

The authors state that previous work provide the agents with total functional, variable, or domain decomposition thereby preventing the agents from making informed local decision about their individual portion of the overall search, in this paper the authors present better method of cooperative mediation that allows the agents to extend and overlap the context that they use for making their local decisions. They referred to (Luo Q et al., 1992) this work present major decomposition technique based on breaking apart the search space by assigning particular domain elements from one or more of the variables to individual agents. One major drawback of this technique is that each of the agents has to know the variables, domains, and constraints for the entire problem. The other work that authors referred to (Yokoo M and Hirayama K, 1996) which present variable decomposition technique that involves assigning each agent one or more variables to manage giving each knowledge of the constraints on their variables.

To solve DisCSPs the authors in this paper have developed a cooperative mediation based negotiation protocol, called Asynchronous Partial Overlay (APO) algorithm, the idea of this algorithm is that agents mediate over conflicts, the context they use to make local decisions overlaps with that of the other agents, the agents gain more context information along with critical paths of constrain graph to improve their decision. To evaluate the proposed algorithm the authors implement APO algorithm and compare the results with AWC algorithm (Yokoo M and Hirayama K, 2000). Both algorithms were tested in partially asynchronous simulation environment called Farm. The conducted experiments were in distributed 3-coloring domain. The authors claim that APO algorithm is both "sound and complete", and performs better than AWC algorithm for both sparse and critical graph-coloring problems. Evaluation of APO algorithm in both additional DisCSP domains (such as SAT) and against additional DisCSP technique will be the future work.

Citation: this paper cited by:

Modi P, Veloso M (2005) *Bumping strategies for the multi-agent agreement problem* In:Proc of the 4th international joint conference on Autonomous agents and multi-agent systems.

[14] ^(Modi P et al. , 2004)

The authors in this paper state that one of the more compelling visions for agents research is the development of “personal assistant agents” that is tasked with making people and organizations more efficient. This paper describes CMRadar, a calendar management system that is significant step towards achieving the enduring vision of assistant agents.

the authors referred to several researchers including themselves (Radar project, 2004) the goal of this large project is to develop a personalized agent that is able to assist its user in a wide range of everyday tasks. The idea as presented in this paper based on CMRadar, CMRadar is an agent with capabilities ranging across the full spectrum of calendar management from natural language processing of incoming scheduling-related emails, to making autonomous scheduling decisions, to negotiating with other users, to user interfacing and visualization. The authors have implemented the initial prototype, the simulated experiments result shows that logging past negotiation costs and using them to predict future costs can be used to improve performance. The authors in this paper claim that CMRadar is the first end-to-end agent for automated calendar management. In future work the authors intend to develop more sophisticated reasoning within each component along the dimensions of the key challenges discussed.

Citation:

Faulring A, Myers B (2005) *Enabling rich human-agent interaction for a calendar-scheduling agent*. Conference on Human Factors in Computing Systems.

[15] *(Modi P, Veloso M, 2004)

Deciding when to reschedule an existing meeting in favor of new meetings. The authors present this problem as a challenging problem since scheduling is distributed, incremental and agents are limited in the information they can exchange. The multi-agents MS problem has been previously investigated by (Sen S and Durfee E,1998) in this work the authors formalize the problem and they have identified a family of negotiation protocols for feasible solution in distributed manner.

The authors in this paper have modeled MS problem as a special sub-class of distributed constraints reasoning called (IL-MAP), the idea of this work is to exploit given model of "scheduling difficulty" with other agents in order to improve MS performance. The specific hypothesis that authors investigate is that an agent can use models of the calendar density of other agents where they assume the calendar density is correlated with the agents rank in an organization. The authors have conducted experimental results comparing rescheduling strategies that use a model of "scheduling difficulty" SD with other agent versus strategy and then evaluate each strategy by averaging measurements over number of runs. The authors claim that their SD model strategy significantly reduces the number of scheduling failures. In addition, the high failures rate caused by uncontrolled cascading of bumps is avoided. In future work the authors are interested in how an agent can automatically learn these models from past negotiation history.

[16] ^(Sen S et al. , 1997)

The problem addressed by the authors that routine information processing that needs in organization can be efficiently automated; to achieve this the authors have developed an agent-based meeting scheduling system that can automate and share information processing tasks of scheduling meeting between a group of users. They have referred to their previous work (Sen and Durfee E, 1991;Sen S and Durfee E, 1994) that was focusing on problem of how an application domain for intelligent surrogate agents can be analyzed. One drawbacks of this work was they did not address many implementation issues like communication medium, user interaction, and use of preferences. While in this paper the authors focusing on representing and using user preferences to categorize acceptable/unacceptable-meeting proposals, and to generate and rank alternative meeting times to propose. The idea of autonomous information processing agents as presented in this paper is the agents honor and follow user preferences. While scheduling meetings, users have preferences for various aspects of the meetings details. As each user might have different preferences for day of week, time of day,..etc., surrogate MS agents have to consider these details while negotiating meetings on behalf of its associated user. To avoid conflicting of user preferences they used technique from voting theory which allows an agent to arrive at consensus choice for meeting times while balancing different user preferences. The authors have implemented distributed meeting scheduling system on a work-station-based computing environment, the programming language used for the implementation is Java, in this system the user interacts with the meeting scheduling system through the user interface. This implementation is being used by a small group of researchers and the results as the authors indicate are encouraging. The authors claim that their autonomous scheduling can approximate the privacy and security concerns of users, and allows for better throughput and better fault tolerance.

Citation: this paper cited by:

Azoulay-Schwartz R, Kraus S (2002) *Negotiation on Data Allocation in Multi-Agent Environments*, Autonomous Agents and Multi-Agent Systems 5, 123–172 – Springer.

[17] ^(Sen S, Durfee E, 1996)

The authors state that most of traditional scheduling methods are batch methods where requirements for a set of tasks are presented to the system, in this paper the authors attempt to develop better approach based on developing and evaluating heuristics by which distributed agent can negotiate scheduling of meetings that arrive over time. They referred to (Davis E and Patterson A, 1975). Where a number of heuristics have been developed that can be used by a single scheduler to schedule a batch of tasks. One drawback of this work is these heuristic methods do not process meeting requests that arrive dynamically over time. The authors in this paper used finite state automata model of contracting agents, from this model identify different forms of conflicts that affect the efficiency of the system, then use search biases as heuristics to avoid conflicts, and cancellation and rescheduling mechanism that resolve conflicts when conflicts avoidance is not possible. The authors reported the results from two different sets of experiments involving two different of groups, the obtained results shows that heuristics search biases schedules meeting quicker than LE or LDD search biases, heuristics search bias can better handle disparities between attendee calendars. The authors claim that the proposed system of distributed contract-based negotiating can be effectively used in manufacturing environment. In future work the authors attempt to address broader class of scheduling problems using techniques that discussed in this paper.

Citation: this paper cited by:

Chun A, Wai H, Wong RY (2003) *Optimizing agent-based meeting scheduling through preference estimation* - Engineering Applications of Artificial Intelligence.

[18] *(Sen S, Durfee E, 1991)

The authors state that previous efforts in developing automated meeting schedulers have met limited success since its based on centralizing control in a single resource allocator, and lack of robustness. To overcome these limitations the authors develop distributed decision-making among the process controlling the separate resources. The authors referred to (Kelly J, Chapanis A, 1982) that shows an experimental studies on how humans handle scheduling and suggest strategies to be used in more efficient, popular automated meeting schedulers.

The proposed approach views meeting scheduling as distributed task where a separate calendar management process is associated with each person in order to increase reliability, the authors in this paper have developed formal model for analyzing alternative heuristic strategies that affect the performance and efficiency of the overall search process. The authors in this paper have tested two prediction strategies: Qualitative prediction and Quantitative prediction, they have shown from the obtained results that quantitative prediction can do better than qualitative in building a workable system. The authors claim that their preliminary experimental have confirmed some of the prediction from their model. In future work the authors intend to develop formal theory to address a wider class of problems of interest both to organization computing system and distributed AI.

Citation: this paper cited by:

Sen S, Haynes T, Arora N (1997) *Satisfying User Preferences While Negotiating Meetings.* Int. Journal of Human-Computer Studies (47).

[19] ^(Tsuruta T, Shintani T, 2000)

The problem addressed by the authors that meeting scheduling is over-constrained and no solution exist that can satisfy all constraints, they are trying to develop new method for scheduling meetings that satisfies as many of the important constraints as possible by formalizing MS problem as Distributes Valued Constraints Satisfaction. To formalize MS problem as the DVCSP, the authors referred to (Lemaitre M and Verfaillie G, 1997) work, since this work propose a formalization of DVCSP and a greedy repair distributed algorithm for solving DVCSP, in this algorithm during an agent turn, other agents must not change their local assignment. While the authors in this paper applied distributed synchronous algorithm to MS problems formalized as DVCSP and some agents can change their local assignments simultaneously.

the idea of this algorithm as presented by authors, is an agent corresponds to each group member, this agent maintains its user calendar and preferences for meetings and acts on behalf of its user in meeting scheduling; users are able to keep information regarding their calendars and preferences private. To evaluate the proposed algorithm the authors have implemented Group Schedule Management System by Java and execute the experiments on PowerMacintosh G3 (Power PC G3 400 GHz), they applied the proposed algorithm to problems that were randomly generated by GSM system, and then measured average computation time until the algorithm found solution, and average of valuation. Then applied the DOC method (Bakker R. et al., 1993) to same problems in order to find an optimal solution, and then compared its results with the results of the proposed algorithm. From their experiments, the authors indicate that the proposed algorithm is cost- effective in comparison to DOC method, can discover a semi-optimal solution to over constrained MS problem in practical time, and determining an optimal solution for MS problem is very expensive. The authors claim that their algorithm is cost-effective in comparison with DOC method and the algorithm is superior.

Citation: this paper cited by:

X Luo, J Ho-man Lee, H Leung, NR Jennings (2003) *Prioritised fuzzy constraint satisfaction problems: axioms, instantiation and validation* - Fuzzy Sets and System.

[20] ^(Yokoo M, Hirayama K, 1998)

The authors mentioned that previous algorithms to solve distributed constraint satisfaction problems (DisCSPs) are neither efficient nor scalable to larger problems. Since they assume each agent has only one local variable. In this work the author intend to develop better algorithm that can handle multiple local variable efficiently based on Asynchronous Weak-Commitment search algorithm. They referred to their previous work (Yokoo M, 1995) in which the priority order of agents is changed dynamically. One limitation of this work as it mentioned by the author is the algorithm assume each agent has only one local variable, and this assumption cannot be satisfied when the local problem of each agent becomes large and complex. The authors in this paper propose new algorithm based on Asynchronous Weak-Commitment search algorithm in which each agent sequentially performs the computation for each variable, and communicates with other agents only when it can find a local solution that satisfies all local constraints. By using this algorithm bad local solution can be modified without forcing other agents to exhaustively search local problems, which lead to decrease the number of interactions among agents. From their experimental evaluations the authors claim that the proposed algorithm is far more efficient than their previous algorithm (Yokoo M, 1995) that uses the prioritization among agents.

Citation:

Modi P, Shen W, Tambe M, Yokoo M (2006) *ADOPT: asynchronous distributed constraint optimization with quality guarantees.* Artificial Intelligence.