Progressive Multi-Agent Negotiation*

Lyndon C. Lee
Intelligent Systems Research
British Telecom Laboratories
Ipswich, IP5 7RE, U.K.
lyndon@info.bt.co.uk

Abstract

This research is concerned with coordination among autonomous agents in *cooperative domains* where, despite conflicting interests, agents can always negotiate to increase their mutual benefits. We introduce a multi-agent negotiation model to model explicitly the negotiation process, and a monotonic negotiation protocol to ensure convergence. Particularly, the model formalizes *progressive negotiation* where negotiation among a group of agents can be divided into a number of sub-negotiations which proceed incrementally. We investigate how the sizes of these sub-negotiations affect the negotiation solution.

Introduction

Much research in distributed artificial intelligence (DAI) is concerned with how agents' interaction can be coordinated. However, good coordination is non-trivial because it requires an efficient coordination mechanism supplied with up-to-date domain and meta information (Durfee & Lesser 1987). Moreover, for autonomous agents which react to the environment driven by their *own* utilities, coordination becomes even more difficult because these agents can have conflicting interests or goals. In such situations, researchers investigate how *rational* agents in certain specific domains can gradually, and mutually accommodate each other via constructive communication – a process also known as *negotiation*.

Despite the extensive DAI literature on negotiation, it has rarely been defined precisely. Moreover, very few have formally and explicitly modeled the process of negotiation. As a result, the impact of different negotiation processes on the solutions is not yet fully understood. Furthermore, much research seems to have assumed that all agents have to negotiate together. This could be a severe limitation when communication and computational capacity are too limited for agents to undergo grand negotiation.

In this paper, we propose a formal negotiation model which not only describes the negotiation process precisely but also allows for *progressive negotiations*. In progressive negotiation, the grand negotiation will be divided into a number of sub-negotiations involving only subsets of agents. After these sub-negotiations, which are conducted in successive stages, have finished, a solution consisting of all the agents will be obtained. On the basis of this model, the effect of different sizes of sub-negotiations on solutions is analyzed.

Related Work in DAI

Earlier work on negotiation in DAI concentrates on building negotiation models based on theories and principles discovered in human negotiation strategies (Sycara 1988; Werkman 1990). Various AI techniques are usually combined with and used to model these strategies in order to improve the efficiency of the negotiation. Some other research attempts to develop novel AI techniques based on the characteristics of human negotiations. They see coordination or multi-agent negotiation as a *distributed search* (Durfee & Moutyomery 1991; Lander 1994; Lesser 1990) in DAI. In this case, various AI search techniques are proposed together with focused and purposeful communication to improve the efficiency of coordination (Conry *et al.* 1991; Lander 1994; Sathi & Fox 1989).

On the other hand, some researchers aim to discover the properties of negotiation under certain formal theoretical frameworks and (quite restricted) assumptions. A large body of work in this camp apply game-theoretical or decision-theoretical tools to study how agents should react in a given specific interaction (Binmore & Dasgupta 1987; Kahan & Rapoport 1984; Zeuthen 1930) or how to design the rules of interaction for autonomous agents (Rosenschein & Zlotkin 1994).

These approaches are complementary in that the

* This work was carried out as part of the author's Ph.D. project at the University of Essex, Colchester, U.K.

emphasis of the first two is on the negotiation processes rather than the outcomes, while the emphasis of the third focuses on the properties and outcomes of negotiations without explicitly modeling the process. As a result, some recent research concentrates on investigating the impact of different negotiation processes on properties or outcomes of negotiations (Hu *et al.* 1994; Wooldridge, Bussmann, & Klosterberg 1996). Our work presented in this paper also takes this approach.

Notation

In this paper, standard set theoretic notation will be used where possible, with the following additional notation. Let S be any arbitrary set and T be any arbitrary sequence, then

- $\mathcal{P}(S)$ is the power set of S;
- Let $\text{seq}(S)$ be a sequence over S, i.e. $\text{seq}(S) = \langle s_1, \ldots \rangle$ where $\forall s_i \in \text{seq}(S) : s_i \in S$, and $\text{Seq}(S)$ be the set of all possible sequences over S;
- $\text{seq}_n(S)$ denotes a n-sequence over S, i.e. $\text{seq}_n(S) = \langle s_1, \ldots, s_n \rangle$ where $\forall s_i \in \text{seq}_n(S) : s_i \in S$, and $\text{Seq}_n(S)$ denotes the set of all possible n-sequences over S;
- $T' \subseteq_s T$ means that T' is an order-preserving subset, i.e. a subsequence, of T.

The Problem Domain

Of the many cooperative domains, we specifically confine our investigation on a subclass of domains where agents' interactions are *superadditive* in game theory terms (Kahan & Rapoport 1984). Generally speaking, superadditive domains are those in which agents always benefit by coordinating their activities. Most task re-distribution domains such as the postman problem proposed by Zlotkin and Rosenschein (Zlotkin & Rosenschein 1989) belong to this class, and here the postman problem was chosen as our domain problem.

Briefly, in the postman problem, every agent (i.e. postman) in the post office is given a set of letters and they are required to deliver these letters to the addresses marked on the envelops. Agents first find out the shortest route which they can visit all the addresses required to deliver those letters, carry out the delivery, and then go back to the post office. The problem is to investigate how these agents should negotiate in order to redistribute their letters such that their individual traveling distances are minimized.

Domain Definitions

Let the set of all possible addresses be V and the set of all possible binary connections between any two addresses be E. The multi-agent postman problem can be formalized as a 3-tuple $\langle G, L, w \rangle$ where $G = G(V, E)$ is a weighted graph representing a map in which every $v \in V$ represents an address, and $e \in E$ represents a route connecting two addresses. Furthermore, there is a special address $v_p \in V$ called the "Post Office."

Let the addresses required to visit by a postman i be $L_i \in \mathcal{P}(V)$. Here $L = \{L_1, \ldots, L_n\} = \text{seq}_n(V)$ is a n-sequence of sets of addresses to which every corresponding agent $i : i \in (1, n)$ is assigned initially to deliver some letters. A weight function $w : E \mapsto \mathrm{R}^+$ is used to represent the distance of any given route. Hence, the "cost" of traveling a particular route e can also be measured in terms of the length of the route, i.e. $w(e)$.

The cost of visiting a set of addresses L_i by any agent i will be $C(L_i) \in \mathrm{R}^+$, i.e. the length of the minimal weight cycle that starts at the post office, visits all the addresses defined by L_i, and ends at the post office. The task in this problem is to redistribute agents' tasks (addresses) so that the cost of every individual agent is minimized. Agent i's utility is defined as $U_i(L'_i) = C(L_i) - C(L'_i)$, i.e. the cost difference between the new set of addresses L'_i (after re-distribution) and the original set L_i.

In this problem, every specific task distribution benefits some agents more than others, i.e. not all agents' costs can be minimized, or not all of their utilities can be maximized. Therefore agents have conflicts to decide how to re-distribute their tasks. Nonetheless, some task distributions do exist to increase agents' utilities, which are preferable to without any re-distribution. In such a situation, we propose that agents negotiate to find gradually an agreement on a particular re-distribution that benefits them all.

A Formal Model of Negotiation

In this section, we present a formal model of multi-agent negotiation called the PEA model for task re-distributed domains. The PEA model is formulated as a utility-driven repetitive process of proposal announcement, proposal evaluation, and proposal adjustment.

Assumptions

- Agents are expected utility maximizers.
- Agents have complete knowledge about other agents' goals and their utility functions.
- Closed world assumption: everything which is not included in the model will not affect the negotiation.
- For any agent, the utility of achieving any subset of its tasks is always less than that of achieving its tasks completely.

- The numbers of agents and tasks are finite.
- Agents are synchronized at discrete time instants.

The Negotiation Process: An Overview

Every agent is assumed to have assigned initially a number of tasks to achieve, and they negotiate among themselves to find out how to re-distribute these tasks to make everyone better off than without negotiation.

In the PEA model, every agent at each time instant chooses and announces a proposal of task re-distribution from their *negotiation sets* to other agents; and the set of all announced proposals is called the *negotiation state*. Proposals are chosen on the basis that they would provide these agents with the greatest *expected utility* at the time. If there is more than one such proposal, agents select one randomly. However, in order to prevent the negotiation from continuing indefinitely, it is defined that if *deadlock* occurs, negotiation terminates and agents must achieve their original tasks individually. Here deadlock occurs if and only if any two identical negotiation states occur.

In this case, a *negotiation protocol* is used to prevent deadlock from occurring by defining a set of proposals *eligible* for the next announcement. By evaluating the last negotiation state, agents can determine their sets of eligible proposals according to the protocol. Whilst the set of eligible proposals is confined by the protocol, the *actual* proposal that an agent will choose to announce depends on its *negotiation strategy.* When a negotiation state occurs in which all proposals are identical in terms of utility distribution, the negotiation ends with any proposal in this state as the solution.

Moreover, instead of negotiate together, agents can also engage into the *progressive negotiation.* Here, negotiating agents are divided into a number of *subgroups*, and *sub-negotiations* take place among agents in each of these subgroups in a consecutive manner. When any sub-negotiation is finished, a *sub-coalition* is formed and will participate in the next sub-negotiation as an individual agent. When a *grand coalition* is formed such that all agents are involved, the progressive negotiation ends.

Definitions

Let $A : A = \langle 1, \ldots, n \rangle$ where $n \geq 2$ denote a sequence of negotiating agents, and agents are assumed to be given a set of tasks to achieve. Let the set of all possible tasks be T, and the set of tasks initially assigned to agent i be $p_{i0} : p_{i0} \in \mathcal{P}(T)$. The set of all possible sub-proposals is therefore denoted by $\mathcal{P} : \mathcal{P} = \mathcal{P}(T)$.

Agents negotiate among themselves by exchanging *proposals* at every discrete time instant. The proposal of i, $P_{i,t}$, which consists of an n-sequence of *sub-proposals* $\langle p_{i1}, \ldots, p_{in} \rangle$, is a proposition of tasks re-distribution that i makes to other agents at time t. Each sub-proposal $p_{ij} : p_{ij} \in \mathcal{P}(T)$ prescribes the tasks that i proposes that j should achieve. Therefore,

$$P_{i,t} = \langle p_{i1}, \ldots, p_{in} \rangle \in \mathrm{Seq}_n(\mathcal{P}).$$

The *utility* of any proposal $P_{j,t}$ to i is defined by

$$U_i(P_{j,t}) = C(p_{i0}) - C(p_{ji}),$$

where C is the cost function that calculates the cost of achieving a set of tasks. Let $\mathcal{P} : \mathcal{P} = \mathrm{Seq}_n(\mathcal{P}) = \mathrm{Seq}_n(\mathcal{P}(T))$ be the set of all possible proposals. Then the *negotiation set* NS_i of i, which consists of all possible proposals, is

$$NS_i = \mathrm{Seq}_n(\mathcal{P}) = \mathrm{Seq}_n(\mathcal{P}(T)).$$

It is assumed that in any NS_i, proposals are organized in decreasing order of i's utilities. However, if many proposals have the same utility to i, they are assumed to be organized *randomly* among themselves.

When agents have received proposals announced by others at t, they are said to be in a particular *negotiation state* S_t. A negotiation state S_t is an n-sequence of proposals announced by agents at t.

$$\mathsf{S}_t = \langle P_{1,t}, \ldots, P_{n,t} \rangle \in \mathrm{Seq}_n(\mathcal{P}).$$

Let $\mathcal{S}$ be the set of all possible negotiation states, so that

$$\mathcal{S} = \mathrm{Seq}_n(\mathcal{P}) = \mathrm{Seq}_n(\mathrm{Seq}_n(\mathcal{P})) = \mathrm{Seq}_n(\mathrm{Seq}_n(\mathcal{P}(T))).$$

If agents are allowed to choose their proposals without any control, the negotiation may proceed ineffectively, or worse still, continue indefinitely or result in conflict. Therefore, unwanted behavior in negotiation is prevented by means of a set of external constraints known as *negotiation protocol* that defines the set of eligible proposals for announcement at any time and state. Formally, a negotiation protocol $\phi : \mathcal{NS}_i \times \mathcal{S} \mapsto \mathcal{P}(\mathcal{P}) \times \mathcal{S}$ is defined as:

$$\phi(NS_i \times \mathsf{S}_{t-1}) = \widehat{NS}_i \times \mathsf{S}_{t-1},$$

where $\widehat{NS}_i \subseteq_s NS_i$ and in which all proposals satisfy the external constraints.

Whilst the protocol determines the set of proposals that *can* be announced in a particular state, among these eligible proposals, the actual one chosen for announcement is left entirely to the agent concerned. Thus, agents use a set of *internal constraints* called *negotiation strategy* to determine the proposals for the next announcement. Formally, a negotiation strategy $\sigma_i : \mathcal{P}(\mathcal{P}) \times \mathcal{S} \mapsto \mathcal{P}$ of any agent i is defined as:

$$\sigma_i(\check{\mathcal{P}} \times S_{t-1}) = \widehat{P},$$

where S_{t-1} is the present negotiation state, $\widehat{P} \in \check{\mathcal{P}} \cup S_{t-1}$ where $\check{\mathcal{P}} \in \mathcal{P}(\mathcal{P})$, and $\widehat{P}$ satisfies the internal constraints.

Here, an agent i's negotiation behavior can be modeled by a *negotiation function* NF_i that always determine proposals for the next announcement according to the protocol and its strategy. So,

$$NF_i \equiv \sigma_i \circ \phi.$$

Moreover, in order to prevent negotiation from proceeding indefinitely, *deadlock* occurs whenever a negotiation state repeats itself at any time. In this case, agents will have to achieve their initial tasks without any task re-distribution. In contrast, when a negotiation state occurs at which all proposals announced are equal in terms of utility distribution, the negotiation ends with any proposal in this state as the solution.

As a result, we can define a negotiation based on the $\mathbb{PEA}$ model as follows:

Definition 1 *A Negotiation is a tuple* $\langle A, \mathsf{NS}, \mathsf{NF}, \mathsf{U}, \phi, \Sigma^n\rangle$ *where* $A = \langle 1, \ldots, n\rangle$ *is a sequence of negotiating agents;* $\mathsf{NS} = \langle NS_{1,1}, \ldots, NS_{n,1}\rangle$ *is the sequence of initial negotiation sets of each agent;* $\mathsf{NF} = \langle NF_1, \ldots, NF_n\rangle$ *is the sequence of negotiation functions of each agent;* $\mathsf{U} = \langle U_1, \ldots, U_n\rangle$ *is the sequence of utility functions of each agent;* ϕ *is a negotiation protocol; and* $\Sigma^n = \langle \Sigma_1, \ldots, \Sigma_n\rangle$ *is the sequence of strategies of each agent.*

Monotonic Negotiation Protocol

In this section, the *monotonic negotiation protocol* (MNP) is introduced. MNP defines the eligibility of proposals for announcement at any time and state, and hence ensures deadlock will never occur.

Definition 2 *A monotonic negotiation protocol is defined by the monotonic negotiation function such that for any times* t, t' *for which* $t > t' \geq 1$, *agent* $i : i \in A$ *has the following options in two different situations:*

- *if* $R_i \neq \min(R_1, \ldots, R_n)$, *then*

$$NF_i(NS_{i,t} \times S_{t-1}) = \begin{cases} P_{i,t} = P^i_{i,t-1} \\ \quad \textit{if } P^i_{i,t-1} \in NS_{i,t}{}^1 \\ P_{i,t} = P^j_{j,t-1} \in S_{t-1} \textit{ only if} \\ \quad U_i(P^j_{j,t-1}) \geq U_i(P_{i,t-1}) \ \& \\ \quad NF_j(NS_{j,t} \times S_{t-1}) = P_{j,t} = P^j_{j,t-1} \ \& \\ \quad U_i(P^j_{j,t-1}) = \max^n_{k=1,k\neq i} U_i(P^k_{k,t-1}) \ \& \\ \quad R_j \neq \min(R_1, \ldots, R_n) \end{cases}$$

- *if* $R_i = \min(R_1, \ldots, R_n)$, *then*

$$\begin{aligned} & NF_i(NS_{i,t} \times S_{t-1}) = \\ & \quad P_{i,t} = P^j_{j,t-1} \in S_{t-1} \textit{ only if} \\ & \qquad U_i(P^j_{j,t-1}) \geq U_i(P_{i,t-1}) \ \& \\ & \qquad NF_j(NS_{j,t} \times S_{t-1}) = P_{j,t} = P^j_{j,t-1} \ \& \\ & \qquad U_i(P^j_{j,t-1}) = \max^n_{k=1,k\neq i} U_i(P^k_{k,t-1}) \ \& \\ & \qquad R_j \neq \min(R_1, \ldots, R_n) \\ & \quad P_{i,t} \in NS_{i,t} \textit{ on the condition that} \\ & \qquad \forall t' < t : \textit{if } P^i_{i,t'} \in S_{t'} \ \& \ P^i_{i,t'} \in NS_{i,t'} \\ & \qquad \textit{then } U_i(P_{i,t}) \leq U_i(P^i_{i,t'}) \ \& \\ & \qquad P_{i,t} \neq P^i_{i,t'} \end{aligned}$$

The above options represent respectively that when i does not have the minimum risk value R_i (see below), i may (1) insist on its present self-proposal; or (2) accept another agent's self-proposal, say $P_{j,t-1}$, when the following conditions are met. First, i's utility of $P_{j,t-1}$ must not be less than its *utility* of its announced proposal $P_{i,t-1}$. Secondly, $P_{j,t-1}$ will be announced by j at t[2]. Moreover, if there is more than one proposal satisfying these conditions, i will accept the one with the *greatest utility*.

In case i has the minimum risk value, then i should either (1) accept another agent's self-proposal when the specified conditions are met; or (2) concede a new proposal from its current negotiation set. In the latter case, i's utility of this new proposal must not be greater than that of its last proposal, and this proposal must never have been announced by i previously.

Risk Function In order that a negotiation does not enter deadlock, at any time there is at least one agent which makes a *concession* by either *accepting* another agent's proposal or *concede* a new proposal. We extend Zeuthen's risk function (Zeuthen 1930) (but due to Harsanyi's formulation (Harsanyi 1977)) to enable agents to determine independently, but coherently, which agents should make the next concession. On the basic of the following multi-agent risk function, those agents which have the minimum risk value should make the next concession.

[1] The new notation $P^j_{i,t-1}$ means that $P_{i,t-1}$ is a proposal that i accepted from j and announced at time $t-1$. On the other hand, $P^i_{i,t-1}$ refers to the proposal $P^i_{i,t-1}$ that i selected from its negotiation set and announced at time $t-1$.

[2] This can be determined by reasoning whether j will accept other proposal (i.e. whether the conditions of option (2) are met) and whether j will announce a new self-proposal (i.e. whether j has minimum risk value). In addition, in order to avoid the circular wait situation when options (2) are met for more than one agent, a tie-breaking algorithm is used (Lee 1996).

$$R_i = \prod_{\substack{j=1 \\ j \neq i}}^{n} r_{i,j};$$

where at any time t,

$$r_{i,j} = \begin{cases} 1 & \text{if } U_i(P_{i,t}) = U_i(P_c)^3 \\ 1 & \text{if } U_i(P_{i,t}) \leq U_i(P_{j,t}) \\ 1 & \text{if } V(P_{i,t}) \geq V(P_{k,t}) \\ & \text{where } 1 \leq k \leq n,\ k \neq i^4 \\ \frac{U_i(P_{i,t}) - U_i(P_{j,t})}{U_i(P_{i,t}) - U_i(P_c)} & \text{if } P_{i,t} \in NS_{i,t} \text{ and} \\ & P_{j,t} \neq P_{i,t-1} \\ 1 & \text{otherwise.} \end{cases}$$

The semantics of the conditional equations in above formulation are as follows. First, individual rationality. An agent will not make any further concession when its utility of its proposal is as great as that of the *conflict proposal* (see [3]). This is because in this case, whether the negotiation is successful or not are indifferent to this agent. The second equation is set because negative risk value has no meaning in the MNP protocol. Third, when any agent has announced a proposal dominating all other proposals, a *pre-solution state*, by definition, has already occurred (see below). Hence no agent is required to make any further concession. The forth equation is due to the risk function proposed by Zeuthen.

Provided that at any time t there is at least one agent i which has announced a proposal $P_{i,t-1}$ from its $NS_{i,t-1}$ and this $P_{i,t-1}$ is then removed from its present $NS_{i,t}$, then the MNP protocol can ensure the negotiation will never enter deadlock. Such removal of a proposal from negotiation set implies that at least one agent will make a concession at every time instant.

Theorem 1 *MNP is deadlock-free if at any time t,*

$$\exists i \in A, P_{i,t} \in \mathsf{S}_t, P_{i,t} \in NS_{i,t} \Rightarrow P_{i,t} \notin NS_{i,t+1}.$$

Proof: For the proof of this theorem and subsequent theorems, see (Lee 1996). □

Whenever a negotiation state occurs in which one or more proposals exist to provide every agent a utility no less than that of their own proposals, by definition, every agent should accept this proposal and hence the negotiation ends. Such negotiation state is thus called the *pre-solution state*. Let $\mathfrak{S}$ be the set of all negotiation states satisfying such condition as follows:

$$\mathfrak{S} = \{\mathsf{S} \mid \forall P_i \in \mathsf{S}, \exists P_j \in \mathsf{S}, U_i(P_i) \leq U_i(P_j)\}.$$

Theorem 2 *If $\forall i,j \in A$, $NS_{i,1} = NS_{j,1}$, $\mathfrak{S}$ is not empty and some $\mathsf{S} \in \mathfrak{S}$ will always occur in finite time t.*

Theorem 3 *In MNP, if $\forall i,j \in A, NS_{i,1} = NS_{j,1}$, then negotiation will always terminate with a solution in finite time.*

Progressive Negotiation

In progressive negotiation, agents in A are first divided into a number of *subgroups* $A' : A' \subseteq A$. For each subgroup, a *sub-negotiation* $\mathbb{SN}_{A'}$, which is an ordinary negotiation using the MNP protocol, will be carried out in consecutive fashion. Whenever a sub-negotiation is finished with a *partial solution* $P_{A'}$, a sub-coalition is formed and will then participate in the next sub-negotiation as if it were an individual agents called *sc-agents*. Hence, more and more agents are involved in sub-negotiation incrementally, and the progressive negotiation ends when a grand coalition containing all the agents is formed.

Without loss of generality, let $A' = \langle 1, \ldots, n' \rangle$ be any subgroup of A where $n' : 2 \leq n' \leq n$. The number of agents n' in A' is called the *subgroup size* of the sub-negotiation $\mathbb{SN}_{A'}$ of A', and the sc-agent of any sub-group A' is denoted by a'.

Definition 3 *A Sub-Negotiation $\mathbb{SN}_{A'}$ is a tuple $\langle A', \mathsf{NS}, \mathsf{NF}, \mathsf{U}, \phi, \Sigma^{n'} \rangle$ where subgroup $A' = \langle 1, \ldots, n' \rangle$ is a sequence of negotiating agents; $\mathsf{NS} = \langle NS_{1,1}, \ldots, NS_{n',1} \rangle$ is the sequence of initial negotiation sets of each agent; $\mathsf{NF} = \langle NF_1, \ldots, NF_{n'} \rangle$ is the sequence of negotiation functions of each agent; $\mathsf{U} = \langle U_1, \ldots, U_{n'} \rangle$ is the sequence of utility functions of each agent; ϕ is a negotiation protocol; and $\Sigma^{n'} = \langle \Sigma_1, \ldots, \Sigma_{n'} \rangle$ is the sequence of strategies of each agent.*

A sub-coalition is formed when a sub-negotiation $\mathbb{SN}_{A'}$ of any subgroup A' has finished. This sub-coalition is then regarded as an individual agent called the *sc-agent* that has the following characteristics:

- The tasks $p_{a'0}$ of an sc-agent a' is the union of the tasks of every agent in the subgroup $A' = \langle 1, \ldots, n' \rangle$.
- The utility function of sc-agent a' is defined as the maximization function of all the utility functions of agents in A'. For any P_i,

$$U_{a'}(P_i) = \max_{p_1 \cup \ldots \cup p_{n'} = p_{ia'}} [U_1(p_1) + \ldots + U_{n'}(p_{n'})],$$

where $p_{ia'} : p_{ia'} \in P_i$ is the set of tasks i proposes that a' should achieve. In other words, the utility

[3] $P_c = \langle p_{10}, \ldots, p_{n0} \rangle$, which consists of agents' initial sets of tasks, is called the conflict proposal.

[4] $V(P_{i,t})$ refers to the *utility vector* of proposal $P_{i,t}$. For any two vectors V, V' of n real numbers, $V \geq V'$ iff $v_i \in V, v'_i \in V', v_i \geq v'_i, \forall i \in (1, n)$.

of P_i to sc-agent a' is calculated by the maximum sum of utility by optimally re-distributing tasks $p_{ia'}$ among agents in A'.

- Sc-agent a' has its own negotiation strategy $\sigma_{a'}$.

Definition 4 *A Progressive Negotiation is a sequence of single sub-negotiations*[5]

$$\langle \mathrm{SN}_{A^1}, \ldots, \mathrm{SN}_{A^m} \rangle.$$

such that

- *All sc-agents formed in any sub-negotiations (except the last one) must participate in the next sub-negotiation:* $\forall k < m, a^k \in A^{k+1}$.
- *A grand coalition must be formed in the last sub-negotiations:* $\forall i \in A$, A^m *contains* i *in such a way that either* $i \in A^m$ *or* $i \in A^k$ *such that* A^m *contains* a^k *for some* k.
- *Any utility obtained by an sc-agent, say* a', *from a sub-negotiation will be proportionally distributed to every individual agent (and sc-agent) in subgroup* A'. *The distribution is based on the utility distribution agreed during the formation of sc-agent* a'. *Let* $P_{A^k}, P_{A^{k-1}}$ *be the solutions of sub-negotiation* $\mathrm{SN}_{A^k}, \mathrm{SN}_{A^{k-1}}$ *respectively, and* $a^{k-1} : a^{k-1} \in A^k$ *be an sc-agent formed by* $\mathrm{SN}_{A^{k-1}}$. *Utility* $U_{a^{k-1}}(P_{A^k})$ *is proportional distributed among agents in* A^{k-1} *based on the following formula:* $\forall i \in A^{k-1}$,

$$U_i(P_{A^k}) = U_{a^{k-1}}(P_{A^k}) \times \frac{U_i(P_{A^{k-1}})}{\sum_{j \in A^{k-1}} U_j(P_{A^{k-1}})}.$$

Analysis of the Progressive Model

In order to understand the relations between subgroup sizes and solution qualities, it is necessary to analyze how the *composition* of the *NS* will change while the group size varies. In the following we shall show that the numbers of efficient and Pareto optimal proposals are in general monotonically increasing as the group size n.

Composition of *NS*

Suppose there are n agents and m distinct tasks. Among all the possible proposals of task distributions, a *efficient proposal* is one where these m tasks are divided into n groups in such a way that the *sum of the utility* to all agents is maximized. Similarly, a *Pareto optimal proposal* is one in which these m tasks are distributed in such a way that it is not *dominated* by any other proposals.

[5] A more general progressive negotiation model can be found in (Lee 1996).

Suppose n is constant. If m increases, the number of possible proposals of task distributions will increase, and so will the number of Pareto optimal (and efficient) proposals. However, empirical observation suggests that the number of optimal (efficient) proposals is at most a polynomial function of the problem size m (Lee 1996).

On the other hand, for a fixed $m : m \geq n$, the number of optimal (efficient) proposals is found also to be a polynomial increasing function of n. Nevertheless, when $m < n$, since some agents will not be allocated any task, therefore the number of optimal (efficient) proposals will generally increase by a factor of n as n rises. Hence, it is hypothesized that the numbers of Pareto optimal and efficient proposals, $P(m,n)$ and $E(m,n)$ respectively, are functions of m and n,

$$P(m,n) = \begin{cases} n^{c_1} \times m^{c_2} & \text{if } m \geq n \\ n \times m^{c_2} & \text{if } m < n; \end{cases}$$

$$E(m,n) = \begin{cases} n^{c_1} \times m^{c_2} & \text{if } m \geq n \\ n \times m^{c_2} & \text{if } m < n, \end{cases}$$

where c_1, c_2 are constants of real number. Since the total number of proposals increases exponentially as m and n, therefore we can deduce that,

Theorem 4 *The proportion of Pareto optimal (efficient) proposals in a complete negotiation set decreases as either n or m or both increase.*

It is axiomatic that the greater is the proportion of optimal (efficient) proposals to be announced, the more likely the solution is optimal (efficient). Based on this axiom, the implication of the above theorem is that the average number of optimal (efficient) solutions is decreasing as m and/or n. Hence, we propose:

Hypothesis 1 *The average percentage of efficient solutions decreases as subgroup size* n *rises.*

Hypothesis 2 *The average percentage of Pareto optimal solutions decreases as subgroup size* n *rises.*

Empirical Results

In order to find out the relations between subgroup sizes and solution qualities, simulation experiments of progressive negotiations based on the formal model using various subgroup sizes were conducted. However, since in practice agents usually cannot produce complete negotiation sets, the complete knowledge assumption is relaxed in experiments, where agents may have only partial view of their *NSs* at any time.

Methodology

In each four-agent (five-agent) experiment, three (four) *sets of simulations* were performed by varying the subgroup sizes from 2 to the grand group size 4 (5). In each

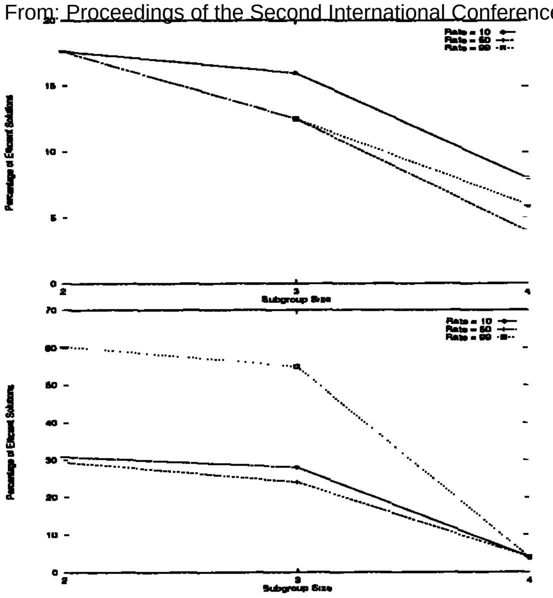

Figure 1: Four-agent Problems

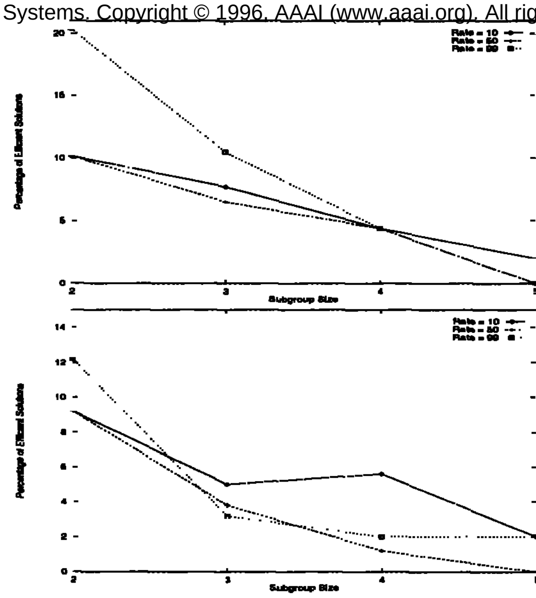

Figure 2: Five-agent Problems

set of simulation of a particular subgroup size, simulations corresponding to all possible agent allocations to subgroups were performed. Moreover, every simulation consists of 50 simulated progressive negotiations, where proposals with the same utility to an agent in its negotiation set were randomly re-organized. Furthermore, the whole set of experiments was repeated using three different concession rates[6]: 10, 50, and 99.

Discussion

Figures 1–4 show the variation of efficient and Pareto optimal solutions as subgroup size rises. Generally, the empirical results are consistent with our prediction in four-agent problems. However, the result of five-agent problem shows that the results of size-4 simulations is greater than expected. For example, the average percentage of Pareto optimal solutions resulting from size-4 simulations is higher than that from size-3, as figure 4 indicates.

It is found that the exception is probably because the "size-4" simulations in fact consists of a mixture of size-4 and size-2 sub-negotiations, contrary to the expectation that it consists purely of size-4 sub-negotiations. The reason for this is that for five-agent progressive negotiation, if the subgroup size of the first sub-negotiation is 4, then the second sub-negotiation must involve two agents: the sc-agent from the first sub-negotiation and the remaining agent. Hence, its subgroup size is 2. Moreover, note that size-2 sub-negotiations, according to our analysis, should produce greater efficient or Pareto optimal solutions than that of size-4 equivalent. Under the circumstances, if the size-2 sub-negotiations have a more dominant effect than the size-4, the overall results from size-4 simulations will then be greater than expected – just as the experiments have shown.

[6]Concession rate r means that an agent will choose the r^{th} proposal in its negotiation set for announcement if all the first r proposals have the same utility as its present proposal (Lee 1996). After announcement, all proposals preceding this chosen one will be removed from its negotiation set. However, if a proposal with less utility is found in the first r proposals, the agent will simply announce it instead.

Conclusion

In order to design agents with sophisticated coordination capability, we must first understand how different designs affect the outcomes of coordination. In the case of negotiation, our investigation on the relations between different negotiation processes and solution qualities provides a starting point toward such understanding.

In addition to efficiency and Pareto optimality, we have also investigated the effect of subgroup sizes on *group utility* and *negotiation time*. Moreover, the impact of different *orders* of sub-negotiations, and of different concession rates on solution qualities have also been analyzed (Lee 1996).

Acknowledgements I would like to thank Jim Doran for his comments on this paper. This research was supported by Croucher Foundation Scholarship.

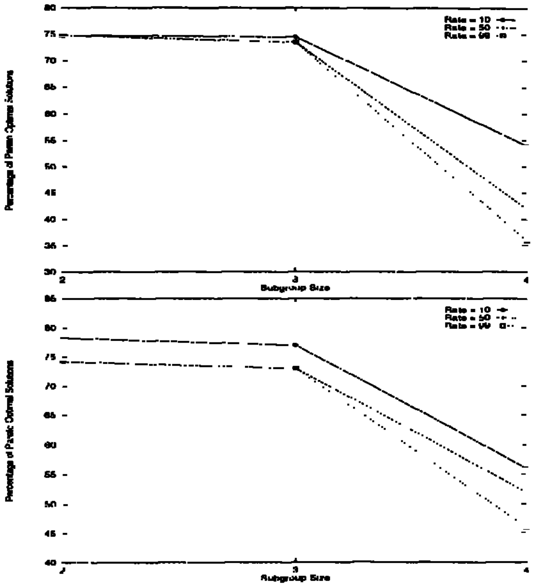

Figure 3: Four-agent Problems

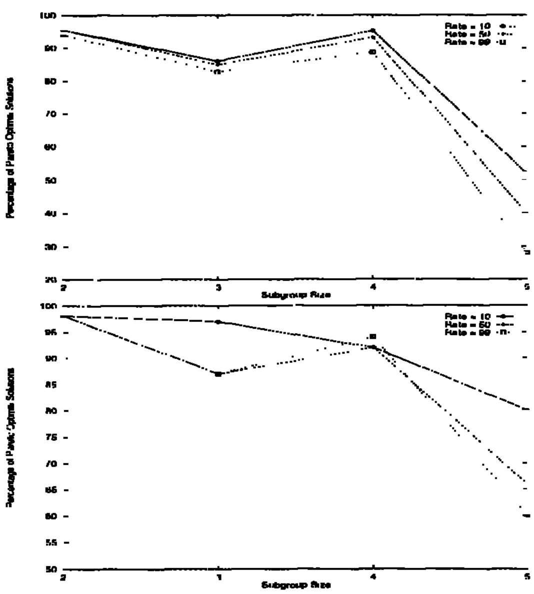

Figure 4: Five-agent Problems

References

Binmore, K., and Dasgupta, P., eds. 1987. *The Economics of Bargaining.* Basil Blackwell.

Conry, S.; Kuwabara, K.; Lesser, V.; and Meyer, R. 1991. Multistage negotiation for distributed constraint satisfaction. *IEEE Trans. SMC* 21(6):1462–1477.

Durfee, E., and Lesser, V. 1987. Using partial global plans to coordinate distributed problem solvers. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 875–883.

Durfee, E., and Montyomery, T. 1991. Coordination as distributed search in a hierarchical behavior space. *IEEE Trans. SMC* 21(6):1363–1378.

Harsanyi, J. 1977. *Rational Behaviour and Bargaining Equilibrium in games and Social Situations.* Cambridge University Press.

Hu, P.; He, Y.; Shi, C.; and Wang, K. 1994. Static negotiation algorithms. In *Proceedings of the First Canadian Workshop on Distributed Artificial Intelligence.*

Kahan, J., and Rapoport, A. 1984. *Theories of Coalition Formation.* Lawrence Erlbaum.

Lander, S. 1994. *Distributed Search and Conflict Management Among Heterogeneous Reusable Agents.* Ph.D. Thesis, University of Massachusetts.

Lee, L. 1996. Negotiation strategies and their effect in a model of multiagent negotiation. Ph.D. Thesis, Department of Computer Science, University of Essex, Colchester, U.K. Forthcoming.

Lesser, V. 1990. An overview of DAI: Viewing distributed AI as distributed search. *Journal of Japanese Society for Artificial Intelligence* 5(4):392–400.

Rosenschein, J., and Zlotkin, G. 1994. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers.* MIT Press.

Sathi, A., and Fox, M. 1989. Constraint-directed negotiation of resource reallocation. In Huhns, M., and Gasser, L., eds., *Distributed Artificial Intelligence 2.* Pitman. chapter 8, 163–193.

Sycara, K. 1988. Resolving goal conflicts via negotiation. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, 245–250.

Werkman, K. 1990. *Multiagent Cooperative Problem Solving Through Negotiation and Sharing of Perspectives.* Ph.D. Thesis, Lehigh University.

Wooldridge, M.; Bussmann, S.; and Klosterberg, M. 1996. Production sequencing as negotiation. In *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, 709–726.

Zeuthen, F. 1930. *Problems of Monopoly and Economic Warfare.* G. Routledge & Sons.

Zlotkin, G., and Rosenschein, J. 1989. Negotiation and task sharing among autonomous agents in cooperative domains. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 912–917.