Negotiation and Its Role in Cooperative Distributed Problem-Solving[1]

Brigitte Lâasri[2] , Hassan Lâasri, and Victor R. Lesser
Computer and Information Science Department
University of Massachusetts

COINS Technical Report 90–39
May 16, 1990

Abstract

Research in cooperative distributed problem-solving (CDPS) considers how problem-solving can be divided among a group of agents and how to dynamically coordinate these agents so as to achieve effective problem-solving. For some CDPS systems, negotiation plays an important role in how agents cooperate. Negotiation among multiple agents involves agents reaching a consensus through the process of conflict detection, propagation, and resolution. We emphasize that negotiation may be a complex and pervasive process that is not only necessary in resolving conflicts occurring in domain problem-solving but also those occurring in control problem-solving. We also show that in the latter case, negotiation is not limited to only control decisions involving the allocation of tasks to agents. We examine the role of negotiation in the different stages of the problem-solving process, namely during the formulation of top-level goals, the selection of active goals, the solving of these goals, the organization of agents, and the coordination of all of these stages. Previous work on negotiation is also discussed with respect to the different stages where negotiation has been used.

[1]This research was, in part, supported by the Office of Naval Research (ONR) under contract number N00014-89-J-1877, by the French research agency: Institut National de Recherche en Informatique et en Automatique (INRIA), and by the French Lorraine County.

[2]Previous research under the maiden name: Maître.

1 Introduction

Cooperative distributed problem-solving (CDPS) is concerned with how a set of agents can work together to solve problems that are beyond their individual capabilities [4]. CDPS assumes each agent is capable of sophisticated problem-solving and can work semi-autonomously. Agents work cooperatively by (1) using their local expertise, resources and information to formulate partial solutions and by (2) integrating their solutions with those of other agents in order to build an acceptable overall solution. The achievement of an acceptable overall solution may involve agents giving up or only partially satisfying some goals if a consistent solution that meets all requirements cannot be found. These requirements can be local to the individual agent(s) solving a goal or non-local, spanning multiple agents.

Some problem decompositions may involve agents with different perspectives working on the same goal or agents working on interdependent goals which share constraints or require common and limited resources for their solution (i.e., explicit or implicit interacting subproblems). Thus, agents during problem-solving may face conflicting solutions to tasks; additionally, some of these tasks may involve making conflicting control decisions on what is the most appropriate problem-solving organization that agents should adhere to, on what goals to focus on, and on which agents should solve these goals and in which order. Conflicts may arise even if the agents try to have some degree of coordination due to their lack of a global view of the problem-solving state, due to the distribution of information and limited communication bandwidth, or due to their heterogeneous expertise. Agents can deal with such conflicts through *negotiation* [1, 3, 7, 12, 2, 13, 5, 6, 8], the process by which the agents act to resolve inconsistent views and to reach agreement on how they should work together in order to cooperate effectively. The research described in this paper attempts to categorize the various styles of negotiation that could occur depending on (1) whether negotiation is an integral part of the problem-solving or not, (2) the particular problem-solving stage where negotiation is required, and (3) the assumptions on which the agents' cooperation is based. Through this work, we emphasize that negotiation may be a complex process that is not only involved in solving the domain problem but also in solving the control problem. An important part of this work is also to show that use of negotiation to solve the allocation of tasks among agents is only one aspect of the control problem in a CDPS system.

In section 2, we describe negotiation as a process of conflict resolution among the agents and show that there is a need of conflict detection and propagation mechanisms. We also show that in fact, conflict detection, propagation, and resolution are based on formulation and exchange of information especially in terms of proposals and critiques. In section 3, we emphasize the problem-solving stages where negotiation can occur and the styles of negotiation that can be involved. In section 4, we analyse some previous work in the field with regard to the style of negotiation used and the problem-solving stages where negotiation is present and show that to date CDPS researchers have only been able to study some of its specific forms. In section 5, we discuss future directions of this research.

2 The Negotiation Process

Negotiation is a process of conflict resolution among the agents due to their *interdependent* activities. It may result (1) in the recognition of an overconstrained solution (there is no mutual acceptable answer), (2) in the generation of a solution completely acceptable to all parties involved, or (3) in the generation of satisfying solutions that are less than optimal but sufficiently satisfy the agents goals. We see negotiation as an integral part of CDPS if agents cooperate through exchange and evaluation of possibly conflicting partial results for performing their local problem-solving as opposed to agents interacting to resolve conflicts only after independently completing their local problem-solving.

2.1 Exchanged information

The negotiation process is based on formulation and exchange of information among the agents especially in terms of *proposals* and *critiques* as described below.

2.1.1 Proposals

By a proposal, we mean a *specific solution*, a *partial result*, a *cluster solution*, or a *partial cluster*. To be more concrete let us for example represent the solution search space as a three-dimensional space (x, y, z). In this example, a specific solution would be a point (x_0, y_0, z_0); a partial result would be, for instance, $x = x_0$ which means that the solution is constrained to be on the plane (x_0, y, z); a cluster solution would be $x = x_0$, $a < y < b$, and $c < z < d$ which provides a range of acceptable solutions; finally, a partial cluster would be $x = x_0$ and $a < y < b$ which means a range of acceptable partial results.

An agent may generate proposals:

1. Independently.

2. Based on proposals generated by other agents.

3. Based on critiques made by other agents of previous proposals.

2.1.2 Critiques

An agent may evaluate a proposal formulated by another agent through a critique which can be composed of:

1. A *positive* part that describes the features of the proposal the agent agrees with. It may include, for example, which parts of the proposal fit with agent's local proposals and constraints.
2. A *negative* part that describes the features of the proposal the agent disagrees with (the conflicting part of the proposal). It may include, for example, violated local constraints of the agent and global or local goals that are no longer achievable.
3. An *explanation* part that provides detailed information about why the agent agrees/disagrees with the proposal.
4. A *counterproposal* part that provides an alternative in response to the received proposal.

A critique may help the agent to understand why the evaluating agent agrees/disagrees with its proposal and provides information indicating what would be an appropriate modification to the proposal.

2.2 Conflict detection, propagation, and resolution

The negotiation process can start either explicitly, when an agent critiques another agent's proposal, or implicitly, when separately generated proposals are inconsistent. In the latter case, conflicts can be detected in one of the following ways:

1. A *centralized* process, in which case a specific agent has a global view of the problem-solving process and is responsible for detecting all potential conflicts; the agent can detect conflicts based on its view of the problem-solving state, on received proposals from other agents, or on previously detected ones.
2. A *distributed* process where multiple agents can be involved. In this case, an agent can detect conflicts that are directly specified in received information from other agents (the agent acquires knowledge about a conflict that has been detected by another agent), deduced based on local activities (the agent detects local conflicts independently of the work

performed by the other agents), recognized based on received proposals/critiques (after evaluating received proposals or analyzing received critiques, the agent detects a conflict with its own perspective), or computed from other received or locally detected conflicts.

Since multiple agents are involved in negotiation or may be involved in linked negotiations (i.e., the agents can be negotiating in the context of different but interrelated goals), the need for propagation of a detected conflict arises: *To which agents does this conflict need to be signaled?* They can be:

1. Determined by the agent that detected the conflict based on its local view of the problem-solving process.

2. Fixed, in which case the agents are determined outside the scope of the system.

3. Determined by one or multiple fixed agents based on the context of the conflict.

Once a conflict or multiple conflicts are detected and possibly propagated, the issue of how to solve them arises: *How are conflicts resolved?* This can be done by:

1. A *centralized* process where a specific agent is responsible for resolving conflicts among multiple agents. This case assumes that the appropriate strategies to resolve conflicts and the criteria for stopping the negotiation process are held by this agent.

2. A *distributed* process where multiple agents can be responsible for coordinating the negotiation process. In this case, the strategies for solving a conflict as well as the criteria for stopping the negotiation process can be either common or local to the agents.

The selection of the agents responsible for detecting conflicts, those responsible for propagating conflicts, and those responsible for resolving these conflicts is a part of a required organization among the agents and is discussed in the following section (cf. local organization).

3 Where Can Negotiation Occur in CDPS?

From a global viewpoint, we have identified three conceptual phases that agents can pass through in order to solve a complex problem in CDPS:

1. *Top-level goals formulation* where top-level goals that characterize the requirements for a solution to the problem are identified.

   This phase is usefull (1) at the beginning of problem-solving to determine which set of top-level goals to first select and (2) when the agents face an overconstrained problem that they cannot completely solve and thus decide to reduce the original problem by giving up or modifying existing top-level goals; another alternative consists in reformulating the problem as a new set of top-level goals.

2. *Active goals determination* where a set of goals is selected to be the agents' common and current focus of attention.

   When there are limited resources to solve all goals, there is a need for the agents to determine which goals to work on next.

3. *Active goals allocation and achievement* where active goals are allocated to agents to be solved.

   The most appropriate agents to solve a goal are determined based on their available resources and on their expertise.

It should be noted that these phases can be incremental processes which can be interleaved during problem-solving and can be going on simultaneously. Intermediate results of one phase can trigger the other phases as shown in figure 1. A top-level goal achievement or the recognition of an overconstrained problem may induce new top-level goals to be formulated or existing ones to be given up or modified (links from phase 3 to phase 1). The formulation of new top-level goals, the deletion or modification of existing ones may change the agents' current focus of attention (links from phase 1 to phase 2). A new goal decomposition that results in additional subgoals being generated, a goal achievement, a goal deletion, or a goal's priority modification may also induce new goals to be active and thus change the focus of attention of the agents (links from phase 3 to phase 2). The selection of an active goal triggers its allocation and achievement (link from phase 2 to phase 3).

In this section, we discuss each of these phases by emphasizing the various types of cooperative problem-solving that may be necessary for performing work associated with each phase and thus indicating where negotiation can occur and what styles of negotiation can be involved.

3.1 What are the top-level goals?

Let us define the problem that agents have to cooperatively solve in terms of a set of top-level goals. A first question arises: *What are the top-level goals that agents have to work on?* Let G_i^1 represent these top-level goals. They can be determined:

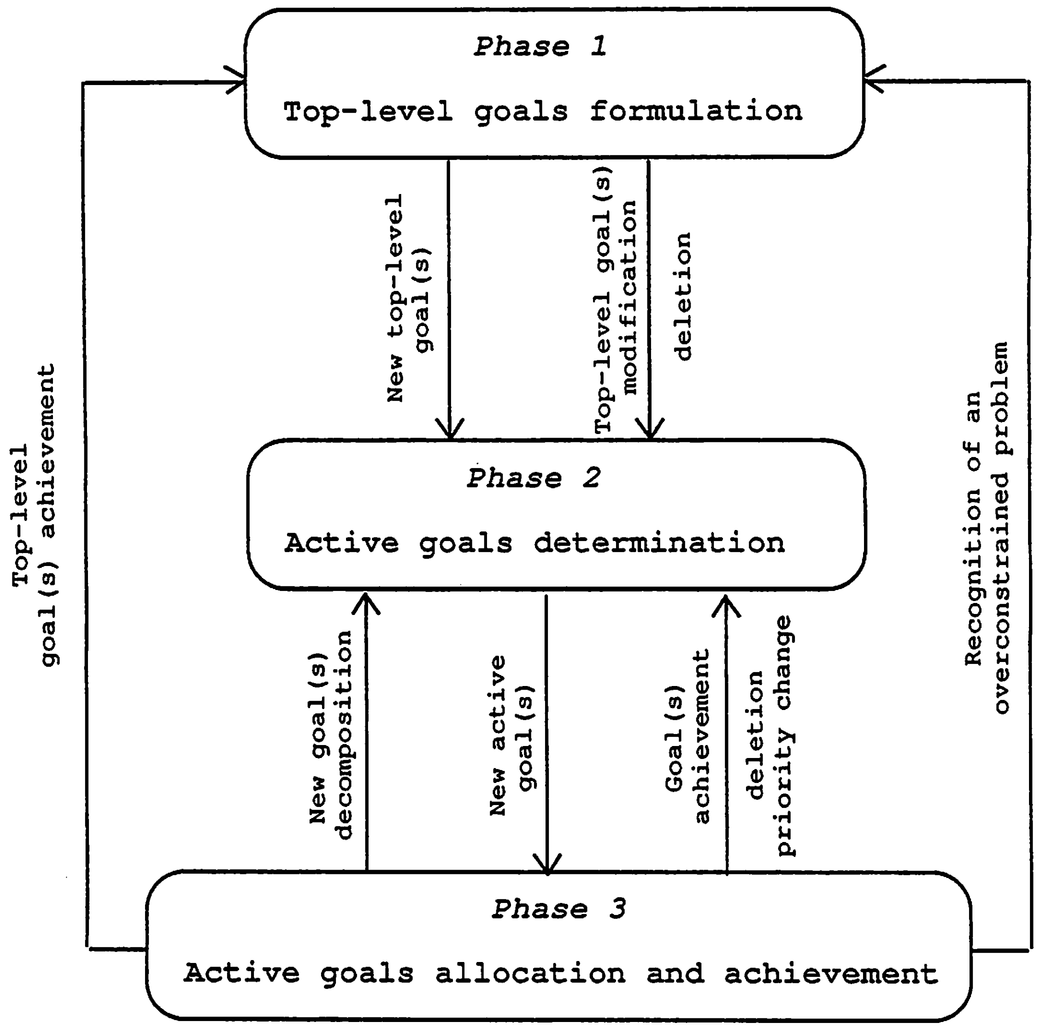

Figure 1: CDPS phases.

1. By a process *outside* the scope of the system: This case refers to CDPS systems where the problem formulation is not a part of the work of the agents. More precisely, it relates to those systems where top-level goals are given a priori to the agents. Such systems can not deal with an overconstrained problem since they have no mechanism that permits them to dynamically formulate another set of top-level goals.

2. By a *non-cooperative distributed* process: This case refers to CDPS systems where the problem formulation phase is broken into *independent* processes, each of which is performed by a fixed agent, and where there is no need of interaction with other agents.

3. By a *centralized* or a *cooperative distributed* process: This case relates to CDPS systems where:

   (a) The problem formulation is performed by an agent which has a global view of the problem-solving and knowledge for generating the top-level goals to work on (centralized).

   (b) Multiple agents are involved in the formulation of top-level goals and where interaction among the agents is required during this phase (cooperative distributed).

   For these two cases, a second question arises: *Which agent(s) will determine G_i^1?* Let A_i^1 represent this/these agent(s). It/they can be determined:

   (i) By a process *outside* the scope of the system: This case refers to CDPS systems where there is a predefined organization of the agents for formulating top-level goals. More precisely, it relates to those systems where this aspect of the organization is static and does not change during the problem-solving process.

   (ii) By a *centralized* or a *cooperative* process: This case refers to CDPS systems where:

   α) The organization of the agents for formulating top-level goals is determined by an agent that has a global view of the problem-solving and knowledge for computing organizations among agents (centralized).

   β) Agents have to organize themselves for defining which of them will participate in the formulation of top-level goals (distributed). In order to select which agent(s) will determine top-level goals, the agents necessarily have to interact. Thus, this process is assumed to be also cooperative.

For these two cases, a third question arises: *Which agents will select agents A_i^1?* Let $A_i^{1\prime}$ represent these agents. We assume that the latter are specified by a process outside the scope of the system[1].

It should be noted that once multiple agents are involved such as in some of the previous cases (agents A_i^1 in case 3b or $A_i^{1\prime}$ in case 3a/3b-iiβ), they have to know which of them will formulate proposals and/or evaluate proposals of others. Thus a *local organization* has to be defined among them in order to respond to the following questions: *how are they organized into agents generating proposals, agents evaluating proposals, and agents generating and evaluating proposals, which agents are responsible for detecting conflicts, which agents propagate conflicts*, and *how are conflicts resolved?* This local organization can be:

- Predefined outside the scope of the system (a role is fixed for each agent).
- Computed by a centralized process based on previous work and on the current problem-solving state.
- Determined by the agents themselves or during their selection.

As shown, during the formulation of the problem, multiple agents may cooperate in defining top-level goals G_i^1 to work on, in selecting agent(s) A_i^1 that will determine these top-level goals, and in determining their local organization.

It is important to note that unless all of the agents have access to the same information and algorithms or work on independent subproblems, each of the previous cases constitutes a source of potential conflicts among the agents and indicates where negotiation can occur during the generation of top-level goals.

3.2 Which goals are active?

Once some top-level goals are known, agents can achieve them either directly or indirectly through decomposition into *common* subgoals. As opposed to "local", by "common" subgoals we mean subgoals that are known by more than one agent and thus can be assigned to other agents to be solved or decomposed. Since there may be insufficient resources to work on all the goals/subgoals that are required to solve the top-level goals, there may need to be a common focus of attention phase where an order for the achievement of goals/subgoals is defined. When common goals (subgoals) are achieved, when existing ones are deleted or have their priority changed, or when new goals

[1]If we suppose that agents $A_i^{1\prime}$ can be determined by a centralized or a cooperative distributed process, then we need some agents $A_i^{1\prime\prime}$ to select $A_i^{1\prime}$, and the need for such higher-level agents can go on indefinitely. To avoid this problem, we assume that at one level, there is some static organization of the agents.

are generated, the agents' current focus of attention can change accordingly. Thus, at each step of the problem-solving process there are some active goals that agents are currently working on. Let G_i^2 represent these goals.

The determination of G_i^2 involves similar cases to those of the previous section. Active goals can be determined by a process outside the scope of the system[2], by a non cooperative distributed process, or by a centralized or a cooperative process. In the latter case, the involved agent(s) A_i^2 can be determined in the same manner as agents A_i^1. Thus, multiple agents may cooperate in selecting these active goals, in selecting agents A_i^2 that will determine these goals, and in determining their local organization. Also, each of these cases may constitute a source of potential conflicts among the agents and indicates where negotiation can occur during the determination of active goals.

3.3 How to allocate and solve active goals?

Since we are concerned with CDPS systems, solving active goals is assumed to be a *cooperative distributed* process where multiple agents are involved. Generally, the agents have various ways to solve active goals directly or indirectly. In fact, there are three basic operations to solve a goal:

1. A goal may be decomposed into common subgoals that can be assigned to other agents to be solved or further decomposed.

2. It may be achieved directly, which means that one or multiple agents are able to tackle it without decomposing it into common subgoals that would have to be assigned to other agents. However, it is possible that the agents decompose the goal locally into internal subgoals in order to achieve it.

3. Since the same goal may be assigned to different agents to be solved through decomposition into subgoals or directly, there is a need for a combine-results operation that permits the synthesis of the agents' various answers.

At this point, a question arises: *For each active goal, which agents will perform which operation?* Let A_i^3, $A_i^{3\prime}$, and $A_i^{3\prime\prime}$ represent the agents that are assigned to directly achieve, decompose, or combine results for that goal, respectively. The three sets can be determined by a process outside the scope of the system (where active goals are predefined and allocated to agents a priori), or by a centralized or a cooperative distributed process in the same manner that agents A_i^1 were determined in section 3.1.

[2]In other words, active goals can be known in advance. This case assumes that top-level goals are predefined. Moreover, if the goal hierarchy is also predefined, the role of the agents will be limited to perform the third phase described in section 3.3.

In summary, one can see that during the determination of how to solve a goal, negotiation may occur (1) to directly achieve, decompose, or combine results of this goal, (2) to select the agents A_i^3, $A_i^{3\prime}$ and $A_i^{3\prime\prime}$, that will perform these operations, and (3) to define local organization among the agents.

3.4 Coordination of the three phases

We saw in the last section the three phases during which negotiation can occur. We noted that these phases can be incremental processes which can be going on simultaneously. If these phases in a CDPS system can in fact be performed simultaneously, then there is a need to coordinate the allocation of resources to each of them. This meta-level resource allocation process has the same issues as how to organize the computation in each phase and thus constitutes a final area that may involve negotiation.

4 An Overview of Related Works

Most related work that has been done on negotiation for CDPS occurs in one narrowly defined phase of problem-solving as summarized in Table 1[3].

	Phase 1		Phase 2		Phase 3 (case 3b)		
	Case	Negotiation	Case	Negotiation	Case	Negotiation Allocation	Negotiation Resolution
Davis & Smith	1	No	2	No	iiß	Distributed	No
Sycara	1	No	1	No	i	No	Centralized
Cammarata et al.	1	No	1	No	i	No	Centralized
Sathi et al.	1	No	1	No	i	No	Distributed
Lander & Lesser	1	No	1	No	i	No	Distributed
Kuwabara & Lesser	3b-iiß	No	1	No	i	No	Distributed

Table 1: Previous approaches with respect to sections 2 and 3.

Davis and Smith [3] studied the use of negotiation for the allocation of tasks in their contract net protocol. At each stage, a task can be directly achieved by an agent without interaction with other agents or decomposed into subtasks to be allocated. In the latter case, the agent interacts with the other agents in order to decide where to transfer its subtasks; this is accomplished

[3]Case values refer to items explicitly listed in section 3.1 but have to be considered according to the associated phase.

by exchanging with them tasks announcements, bids, and awards. They reach an agreement when the manager (which requested bids for solving one of its subtasks) awards a bidder (which proposed bids for solving this subtask) and establishes a contract with it.

In this approach, initial tasks that characterize the problem are predefined and allocated to an agent (case 1, phase 1). A task is active when the agent that generated this task through decomposition decides to allocate it to other agents (case 2, phase 2). Where tasks are allocated is decided by the agents (case iiβ, phase 3) through a negotiation process that is reduced to a mutual selection between the manager and a potential contractor. Agents first make bids and then the manager selects the most appropriate through awards. It should be noted that negotiation occurs only when conflicts arise – multiple bidders propose simultaneously to perform the same task – and occurs between the manager and each of the bidders but not among the bidders.

Sycara [13] built a centralized planner for resolving conflicts between agents' goals. Given a particular conflict and the context in which the conflict occurs, the planner has two alternatives: (1) find a new compromise by using case-based reasoning or multi-attribute preference analysis strategies, or, if some agents disagree, (2) use persuasive arguments to convince them of the proposed compromise by using explanation-based reasoning or try to improve the compromise by asking for justification of disagreements. Her techniques for forming compromises and for maintaining a history of previous negotiations and compromises are useful for taking advantage of the experience the system has acquired in negotiation.

In this approach, the formulation of top-level goals is predefined and allocated to agents (case 1, phase 1). All common goals are active at each step of the problem-solving process (case 1, phase 2) and the agents that will participate in solving these goals through negotiation are predefined (case i, phase 3). The negotiation process is coordinated by the centralized planner which acts as a fixed agent that has knowledge for performing this task. Since conflicts are given as a part of the original problem, there is no conflict detection mechanism.

Cammarata and her colleagues [1] applied negotiation in the domain of distributed air-traffic control where each airplane (agent) has to construct a flight plan that will maintain an appropriate separation from other airplanes and will satisfy other constraints such as getting to the desired destination with minimum fuel consumption. Their most well-developed approach to this problem is a policy called task centralization. In this policy, airplanes involved in potential conflict situations (which occurs when airplanes could become too close, based on their current headings) choose one of the airplanes involved in the conflict in order to resolve it. This airplane acts as a centralized planner to develop a multi-agent plan that specifies the concurrent actions of all the airplanes[4].

The task-centralized approach falls into the same category as the previous one (case 1, phase 1; case 1, phase 2; case i, phase 3) except that the centralized planner is determined by the agents in conflict. However, this selection process doesn't involve negotiation; the agents use the same information and criteria to select the most appropriate one among them for solving the conflicts.

Sathi and his colleagues [12] proposed a negotiation framework based on constraints relaxation for resolving conflicts in project management. They defined a set of strategies, that characterize the way in which negotiation will proceed as a set of negotiation operators such as log-rolling, substitution, cost cutting, bridging, unlinking, mediation, and arbitration that may be applied to a particular conflict.

This approach falls into the same category as the two previous ones (case 1, phase 1; case 1, phase 2; case i, phase 3) except that negotiation is distributed and conflicts arise when agents' local constraints are violated. The strategies used for negotiation are common to the agents implicated in the conflict. In the same manner, Lander and Lesser [10, 11] in their work on cooperating experts propose a framework for conflict resolution between agents where the strategies are compromise, convince, generate alternate proposals randomly, generate alternate proposals by incorporating other agents constraints, generate alternate proposals by looking for alternate ways of fulfilling goals, and revise and merge goals.

In the extended protocol for multi-stage negotiation [9], Kuwabara and Lesser improved the framework proposed by Conry, Meyer and Lesser [2] for detection and resolution of resource allocation conflicts in a distributed network. They focused on determining impact of local decisions on more global decisions by allowing each agent to exchange information on their local resource allocation, plans they cannot pursue for solving other goals according to this choice and information they received from other agents. This framework allows agents to detect top-level goals that cannot be achieved simultaneously and allows them to choose the ones that have to be given up (case 3a/3b-iiβ, phase 1). Since the agents are assumed to have the same view of such an overconstrained situation, this choice isn't negotiated. All goals are active (case 1, phase 2). The allocation of goals to agents predefined (case i, phase 3) and negotiation occurs only to resolve conflicts among agents' local solutions.

[4]Another policy called task sharing was composed of 2 steps; it consisted of choosing the least spacially constrained airplane's flight plan for modification and choosing the most knowledgeable airplane to perform the replanning. This policy was not further evaluated in testbed experiments.

The approaches to negotiation outlined above only dealt with conflicts occurring during one stage of the problem-solving process, namely during the phase 3. These conflicts most often occurred during the solution of active goals (i.e., domain problem) except in the contract net protocol; in this case, negotiation was used to resolve conflicts in the allocation of active goals to agents. Note that none of these approaches discussed the issues of meta-level resource allocation negotiation defined in section 3.4. In fact, all of these approaches were developed with strong implicit assumptions that hid the complexity of the negotiation behavior needed in CDPS.

5 Conclusion and Further Research

We presented in this paper our preliminary efforts to understand from a general perspective the role of negotiation in CDPS. We focused on the negotiation behavior and viewed the need of processes of conflict detection, propagation, and resolution through the exchange of proposals and critiques. Then, we saw that negotiation – whether considered as an integral part of the problem-solving process or not – can occur at each stage of the problem-solving process depending on assumptions related to the system organization and the complexity of the CDPS. In fact, negotiation is useful in CDPS only when the decision process is neither (1) outside the scope of the system, (2) centralized, (3) distributed with all agents having access to the same information and algorithms, or (4) distributed with independent subproblems where there is no need of results synthesis. In addition, negotiation may be used not only for solving the domain problem but also for solving the control problem, in which case it may be a more sophisticated process than allocating tasks to agents.

We also presented an overview of previous work on negotiation and outlined the assumptions and the styles of negotiation involved.

The studies presented in this paper outlined the features required by a generic approach to CDPS through negotiation. At present, we are studying what a generic protocol of negotiation should be in order to carry out all of these capabilities. However, there are some still open questions not discussed in this paper. For example, how does the negotiation complexity increase with the number of participating agents and what is the implication of the system's ability to build and use a history during negotiation.

References

[1] S. Cammarata, D. McArthur, and R. Steeb, "Strategies of Cooperation in Distributed Problem-Solving", *Proceedings of the Eighth International*

Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, 1983, pp. 767-770.

[2] S. E. Conry, R. A. Meyer, and V. R. Lesser, "Multi-Stage Negotiation in Distributed Planning", in A. Bond and L. Gasser, editors, *Reading in Distributed Artificial Intelligence*, Morgan Kaufmann Publishers, 1988, pp. 367-384.

[3] R. Davis and R. G. Smith, "Negotiation As a Metaphor for Distributed Problem-Solving", *Artificial Intelligence*, Vol. 20, 1983, pp. 63-109.

[4] E. H. Durfee, V. R. Lesser, and D. D. Corkill, "Cooperative Distributed Problem-Solving", in A. Barr, P. R. Cohen, and E. A. Feigenbaum, editors, *The Handbook of Artificial Intelligence*, Vol. 4, Addison Wesley, 1989, pp. 83-147.

[5] E. H. Durfee and V. R. Lesser, "Negotiation Task Decomposition and Allocation Using Partial Global Planning", in L. Gasser and M. N. Huhns, editors, *Distributed Artificial Intelligence*, Vol. 2, Pitman, 1989, pp. 229-243.

[6] L. Gasser, N. Rouquette, R. W. Hill, and J. Lieb, "Representing and Using Organizational Knowledge in DAI Systems", in L. Gasser and M. N. Huhns, editors, *Distributed Artificial Intelligence*, Vol. 2, Pitman, 1989, pp. 55-78.

[7] C. Hewitt, "Offices Are Open Systems", *Communications of the ACM*, Vol. 4, No. 3, 1986, pp. 271-287.

[8] C. Hewitt, "What Is Open Systems Science?", *Proceedings of the Ninth Workshop on Distributed Artificial Intelligence*, Rosario Resort, Eastsound, Washington, September 12-14, 1989, pp. 379-386.

[9] K. Kuwabara and V. R. Lesser, "Extended Protocol for Multi-Stage Negotiation", in *Proceedings of the Ninth Workshop on Distributed Artificial Intelligence*, Rosario Resort, Eastsound, Washington, September 12-14, 1989, pp. 129-161.

[10] S. E. Lander and V. R. Lesser, "A Framework for the Integration of Cooperative Knowledge-Based Systems", in *Proceedings of the IEEE International Symposium on Intelligent Control*, Albany, New York, September 1989, pp. 472-477.

[11] S. E. Lander and V. R. Lesser, *Personal Communication*, November 1989.

[12] A. Sathi, T. E. Morton, and S. Roth, "Callisto: An Intelligent Project Management System", *AI Magazine*, Vol. 7, No. 5, Winter 1986, pp. 34-57..

[13] K. Sycara, "Resolving Goal Conflicts via Negotiation", *Proceedings of the Seventh National Conference on Artificial Intelligence*, St. Paul, Minnesota, August 1988, pp. 245-250.