OAR: A Formal Framework for Multi-Agent Negotiation

Jiaying Shen and **Ingo Weber** and **Victor Lesser**
Department of Computer Science
University of Massachusetts
Amherst, MA 01003-4610
{jyshen, imweber, lesser}@cs.umass.edu

Abstract

In Multi-Agent systems, agents often need to make decisions about how to interact with each other when negotiating over task allocation. In this paper, we present OAR, a formal framework to address the question of how the agents should interact in an evolving environment in order to achieve their different goals. The traditional categorization of self-interested and cooperative agents is unified by adopting a utility view. We illustrate mathematically that the degree of cooperativeness of an agent and the degree of its self-directness are not directly related. We also show how OAR can be used to evaluate different negotiation strategies and to develop distributed mechanisms that optimize the performance dynamically. This research demonstrates that sophisticated probabilistic modeling can be used to understand the behaviors of a system with complex agent interactions.

Introduction

In Multi-Agent systems, agents often need to make decisions about how to interact with each other when negotiating over task allocation. Traditionally, research on negotiation is categorized into two general classes: cooperative negotiation and competitive negotiation. In competitive negotiation, agents are self-interested and negotiate to maximize their expected local reward. In cooperative negotiation, agents work to increase the system's social utility. Recent experimental work (Zhang, Lesser, & Wagner 2003; Jung, Tambe, & Kulkarni 2001) found that it is not always beneficial for an agent to cooperate with other agents about non-local tasks even if its goal is to achieve higher social utility. Similarly, if an agent is interested only in its own local reward, sometimes it still should choose to commit to non-local task for other agents instead of its local task. At the same time, researchers look for effective mechanisms to improve social utility even in competitive negotiation (Braynov & Sandholm 2002; Sen 2002). In a complex distributed system, the environment evolves over time. It is virtually impossible for the agents to always obtain and process all the necessary non-local information in order to achieve optimal performance, whether their goals are to maximize the social utility or local reward only. Formally understanding complex behaviors in multi-agent negotiation is very important for designing appropriate mechanisms to achieve optimal performance.

(Shen, Zhang, & Lesser 2004) builds a statistical model for a small cooperative multi-agent system and introduces *attitude parameter* as an effective local mechanism to improve system performance. In this paper, we extend this research and present OAR, a formal framework to study different issues in multi-agent negotiation. There are three components in OAR. **O**bjective functions specify different goals of the agents involved. **A**ttitude parameters reflect the negotiation attitude of each agent towards another agent. **R**eward splitting specifies how a contractor agent divides the reward received for finishing the task among itself and the agents who finish the subtasks. The traditional categorization of self-interested and cooperative agents is unified by adopting a utility view. Both attitude parameters and reward splitting can be used as effective local mechanisms for the agents to realize their goals. We show that OAR can be used to evaluate different negotiation strategies. The closed form statistical analysis presented in (Shen, Zhang, & Lesser 2004) is extended to mathematically analyze the interaction between attitude parameters and reward splitting and their relationship with different objective functions. To our knowledge, no work has been done that formally analyzes the interaction among different negotiation parameters.

An agent is *completely self-directed* when it does not take into consideration how much utility the other agent can potentially gain if it commits to the requested task. In contrast, an agent is *completely externally-directed* if it sees the other agent's gain as its own when negotiating. In OAR, we distinguish the notion of "self-interested" versus "cooperative" from "self-directed" versus "externally-directed". We call an agent *self-interested* if its local goal is to maximize only its local utility and an agent is *cooperative* if it is intent on maximizing the final social utility. The degree of cooperativeness illustrates the goal of an agent, while self-directness is the local mechanism used to achieve the goal. In OAR, we represent them separately with objective functions and attitude parameters and make this distinction explicit.

Using OAR and the extended formal model built for a typical negotiation system, we are able to show:

- The degree of cooperativeness and that of self-directedness are not directly related to each other;
- Reward splitting is needed in addition to attitude param-

eters as a local mechanism to further improve the performance, in both cooperative and non-cooperative systems;

- An agent has to dynamically adjust its local negotiation parameters, i.e., attitude parameters and reward splitting, in order to achieve optimal performance.

Research in Multi-Agent Systems community has been largely heuristic and experimental. (Sen & Durfee 1998) modeled a meeting scheduling problem and is one of the first formal studies of a multi-agent system. Most formal work on negotiation is done in systems with self-interested agents (Sandholm, Sikka, & Norden 1999; Sen 2002; Saha, Sen, & Dutta 2003). (Decker & Lesser 1993) analyzes the need for meta level communication in constructing a dynamic organizational structure. (Vidal 2004) studies the benefits of teaming and selflessness when using multi-agent search to solve task-oriented problems. Work on dynamic coalition formation studies the problem of finding a payoff distribution for a given game and coalition structure such that no agent has incentive to leave the coalition (Klusch & Gerber 2002). This is similar to the reward splitting issue we study in OAR. The difference is that the agents in the coalition find a payoff distribution through negotiation, while the reward splitting in OAR is a local mechanism and it is decided by the contractor agent locally to better achieve its goal in the current environment. (Levine 1998) introduced a model in which agents' utilities are linear in their own monetary income and their opponents', controlled by a parameter called *altruism coefficient*. This is similar to the calculations of both the objective function and the virtual utility in the OAR framework. However, their model is studied in a competitive setting. Neither does it make the distinction between the goal of an agent and the mechanism that an agent may employ to realize its goal. In OAR, we make this distinction clear by controlling these two related but distinct concepts with two different parameters: the objective parameter and the attitude parameter. We demonstrate that this clear distinction is important and necessary. Additionally, OAR enables us to study agents with different organizational goals in a unified setting by simply varying their objective parameters. The uniqueness of OAR lies in the fact that it represents an agent's goal and its local negotiation mechanisms formally, which allows us to model different multi-agent systems with different negotiation protocols in this framework and understand their performance in various environments.

General Problem

Let us formally define the class of problems we study.

There are a group of agents $A_1, A_2, \ldots, A_n$ and a set of tasks $T_1, T_2, \ldots, T_t$. Each task has a number of parameters that observe a distribution:

- r_i: task T_i arrives at time t with a probability of $1/r_i$.
- e_i: the difference between the arrival time of a task T_i and its earliest start time est_i.
- dur_i: the duration of the task T_i.
- sl_i: the difference between the earliest possible finish time of a task T_i and the deadline dl_i.
- R_i: the reward of a task T_i if it's finished.

The relationship of e_i, est_i, dur_i, sl_i and dl_i is illustrated in Figure 1.

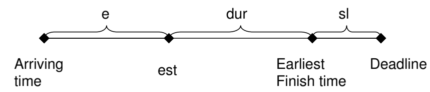


Figure 1: The relationship of different parameters of a task

Each task $T_i, 1 \leq i \leq t$ can be decomposed into a set of subtasks: $T_{i1}, T_{i2}, \ldots, T_{im_i}$, where m_i is the number of subtasks of T_i. All of the subtasks need to be completed in order for the agent A_s at whom T_i arrives to collect the reward. The agent can contract out some or all of the subtasks to other agents or it can finish the task on its own. As a special case, A_s can contract out the entire task T_i. Each subtask $T_{ij}, 1 \leq i \leq t, 1 \leq j \leq m_i$ has a set of parameters as well, and they have to observe certain relationships with each other and with the original task T_i. r_{ij}, e_{ij}, dur_{ij} and sl_{ij} are similar to those of T_i. R_{ij} is the reward of the subtask T_{ij} if it is finished. $\sum_j R_{ij} + R_{i0} = R_i$, where R_{i0} is the reward A_s gets after handing out the rewards to each subtask if all of the subtasks are completed. This reward assignment for subtasks can be decided by either the system or A_s, the contractor of the task.

For each subtask T_{ij} there is a set of agents AS_{ij} who can perform T_{ij}. When a task T_i arrives at agent A_s, A_s starts to negotiate with one of the agent(s) in AS_{ij} for each subtask T_{ij} and transmit the related parameters. When an agent A_l receives a request from agent A_s to do subtask T_{ij}, it decides whether T_{ij} can be fit onto its own schedule or can be contracted out (contracting out a subtask follows the afore mentioned procedure of a regular task). If yes, it commits to the subtask. If there is a conflict between T_{ij} and A_l's own schedule and A_l cannot subcontract T_{ij} out to other agents, it compares the values of the conflicting tasks and commits to the one with highest value, decommitting from the other.

There are three important questions that we need to answer for such a system:

1. What is each agent's goal in the system? Does it want to optimize its local reward, or the total global reward, or a combination of the two?
2. If there is a conflict between tasks, how does an agent evaluate each task and decide which to commit to?
3. When an agent needs to contract out a task, how does it split the reward between itself and the subtasks?

The next section introduces OAR, a formal framework designed to answer these questions.

OAR: The Framework

There are three components in OAR: *Objective functions* that specify the agents' goal, *Attitude parameters* that determine how an agent values each task, and *Reward splitting* that decides the reward allocation of a task that needs cooperation among agents. The closed form statistical model developed in (Shen, Zhang, & Lesser 2004) is extended to incorporate both objective functions and reward splitting in addition to the attitude parameters (Weber, Shen, & Lesser 2005). Due to the space limitation, we do not present the mathematical details of this model, but only discuss the analysis shown in optimality graphs plotted based on this model.

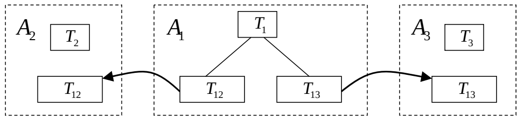


Figure 2: An example three agent organization structure with the necessary inter-agent interactions

Please refer to the Appendix for the formulas used to generate these graphs.

An Example System

Before introducing OAR, we describe a three agent system with the necessary inter-agent interactions during the negotiation process (shown in Figure 2) that exemplifies the class of problems. Being a simple system, it is sufficient to illustrate the complex interactions among the agents. We will use this as an example throughout the rest of the paper.

There are three agents in the system. A_1 has one type of task T_1 with a reward of R_1 and an arrival rate of r_1 coming in, of which there are two different subtasks T_{12} and T_{13} that need to be contracted out to A_2 and A_3 respectively. Suppose at the same time, tasks T_2 and T_3 arrive at agents A_2 and A_3 with rewards of R_2 and R_3 and arrival rates of r_2 and r_3. As a result, there may be conflicts between T_2 and T_{12}, or between T_3 and T_{13}, which force the agents to choose one task between the two. Upon completion of both T_{12} and T_{13}, A_1, A_2 and A_3 each collects a reward of R_{11}, R_{12} and R_{13} respectively.

Objective Functions

Traditionally, research on negotiation is categorized into two general classes: cooperative negotiation and competitive negotiation. In competitive negotiation, agents are self-interested and negotiate to maximize their expected local reward. In cooperative negotiation, agents work to find a solution that increases the sum of the expected reward of all involved agents. However there are other types of agents, whose goal is instead to reach a balance between its local gain and the reward of the rest of the system.

The first component of OAR is *objective function*. It specifies the goal of each agent:

$$O_i = w_i \cdot ER_i + (1 - w_i) \sum_{j \neq i} ER_j, \qquad (1)$$

where ER_i is the expected reward of agent i. $w_i \in [0, 1]$ is called *objective parameter* and reflects how important its local reward is to A_i as compared to that received by the rest of the system. For a fully cooperative agent, $w_i = 1/2$. Its goal is to maximize the expected social utility of the entire system, i.e., $\sum_i ER_i$. A completely self-interested agent is interested only in its own expected reward, and $w_i = 1$. An agent with $w_i = 0$ is altruistic and considers the gains of the other agents only. The objective function unites the traditionally separate views of cooperative systems and a self-interested systems. By simply varying the objective parameter, we can study agents with different goals.

If all the agents are cooperative, the system is a cooperative system, and it achieves its optimal performance when the social utility is maximized. If at least some of the agents are not cooperative, the system is non-cooperative, and a Pareto Nash equilibrium needs to be found.

Attitude Parameters

When an agent receives a subtask proposal from another agent that conflicts with another potential task, it needs to evaluate the subtask in order to decide which task to commit to. Oftentimes it not only cares about the real reward for doing the task but also the reward the rest of the system may gain from it. (Zhang, Lesser, & Wagner 2003) introduced an integrative negotiation mechanism which enables agents to interact over a spectrum of different local cooperation degrees. There are different degrees of local cooperation when an agent is considering whether to cooperate with other agents on an external task (Shen, Zhang, & Lesser 2004). An agent is *completely self-directed* when it does not take into consideration how much utility the other agent can potentially gain if it commits to the requested task. In contrast, an agent is *completely externally-directed* if it sees the other agent's gain as its own when negotiating.

Let us take Figure 2 for example. There are two types of rewards that are transferred from agent A_1 to agent A_2 with the successful accomplishment of task t: real reward R_{12} and relational reward Rr_{12}. Real reward R_{12} has positive benefits to agent A_2. The agent collects real reward for its own utility increase and is calculated into the social welfare increase as well. In contrast, the relational reward Rr_{12} does not contribute to agent A_2's actual utility increase, and is not included in the social utility computation. Instead, it is transferred to reflect how important task t is for agent A_1 and makes it possible for agent A_2 to consider A_1's utility increase when it makes its negotiation decision. Relational reward is a form of meta-information transferred between agents to estimate how an agent's action may affect the rest of the system. In this work, we will not be concerned with lying with respect to Rr. How Rr_{12} is mapped into agent A_2's virtual utility depends on agent A_2's negotiation attitude towards task t with agent A_1.

The second component of OAR is *attitude parameter* k, $0 \leq k \leq 1$. It specifies the negotiation attitude of each agent towards another agent. For a *completely externally-directed* agent, $k = 1$, while $k = 0$ for a *completely self-directed* agent. When an agent A_i receives a task from A_j with a real reward of R and a relational reward of Rr, it evaluates the task by calculating the virtual utility as: $Rn = R + k \cdot Rr$.

In the previous example, A_2 has an attitude parameter k_2 towards A_1. During its negotiation session with agent A_1 about task t, agent A_2 calculates its virtual utility for the task as $Rn_2 = R_{12} + k_2 \cdot (Rr_{12})$ and uses Rn_2 to compare t against conflicting tasks, if any.

Attitude parameter and objective parameter are related but different. They are both used as weight parameters measuring an agent's regard for the other agents' reward in comparison with its own in the calculations of its objective function and a potential task's virtual utility, respectively. However, the objective function and the virtual utility are used in different contexts. The objective function illustrates the goal of an agent and is not a part of the decision process, while the virtual utility is calculated as the expected return of a potential task and is a key component of the local decision process. The agent uses the virtual utilities of conflicting tasks to decide which task to commit to and which to deny. The objective parameter is often set by the system, and usually does not change over time constantly. In contrast, the attitude parameters is a local negotiation mechanism that an agent dynamically adjusts in response to the change in the environment in order to maximize its objective function.

In a complex distributed system, where the environment is evolving over time, an agent does not know all the information about its environment and the other agents. It often has to estimate the value of its objective function. It needs to dynamically choose the level of local cooperation, i.e., the attitude parameter, that is optimal for its organizational goals based on its limited local vision and the information provided by other agents. Experimental work showed that it is not always beneficial for the agents in a cooperative system to be completely externally-directed (Zhang, Lesser, & Wagner 2003). When the uncertainty associated with the utility increase is high, it is better for the agent to be more self-directed. This indicates that complete local cooperation does not always lead to optimal global cooperation. Similarly, the optimal behavior for a self-interested agent is not necessarily always completely self-directed. (Shen, Zhang, & Lesser 2004) demonstrates that appropriate adjustment of attitude parameters is an effective local mechanism to deal with uncertainties caused by the interaction among the agents and the change in the environment.

Reward Splitting

When an agent needs to subcontract several subtasks to other agents in order to finish a task, it is this agent's decision how much to offer the other agents for accomplishing their subtasks. We call this issue *reward splitting*.

In our example system, the shared task T_1 is imposed on A_1, who then negotiates with A_2 and A_3 over the subtasks T_{12} and T_{13}. Agent A_1 has to figure out how much of the overall reward R_1 to pay to the other agents. The rewards for the subtasks are R_{12} and R_{13}, and A_1 keeps R_{11} for itself.

It is a commonly used technique to use real reward as an incentive to manipulate the decisions of other agents. Similar to the use of attitude parameters as a mechanism for the contractee agents to achieve their goals, the reward splitting can be used as a local mechanism for the contractor agents to improve its performance, and is the third component of the OAR framework.

The additional flexibility introduced by reward splitting is necessary to further improve an agent's performance, especially if its goal is not to maximize the social utility. We have seen up to 36% performance difference between different reward splitting settings for non-cooperative agents.

Relational Reward

By taking into account the relational reward instead of just the local reward, an agent that is requested to do a subtask bases its decision whether to do the task not solely on the real reward it may receive, but also on the other agents' rewards. We examine three different ways to calculate the relational reward and their expressiveness.

In the example, A_1 needs to complete task T_1 in order to collect a reward of R_1 and asks A_2 to complete one of its subtasks. It promises a real reward of R_{12} and a relational reward of Rr_{12}. If all the subtasks of T_1 are finished successfully, A_1 itself can collect R_{11} after handing out all the real rewards. Rr_{12} can be calculated in one of the following three ways:

- $Rr_{12}^{(1)} = \frac{1}{2}R_{11}$;
- $Rr_{12}^{(2)} = R_{11}$;
- $Rr_{12}^{(3)} = R_1 - R_{12}$.

Each of the three calculations have its own motivation. Using $Rr_{12}^{(2)}$ as the relational reward, A_2 considers the reward A_1 may receive if it commits to the subtask. The motivation behind $Rr_{12}^{(1)}$ is that A_2 alone committing to the task is not going to get A_1 the reward, but only with some probability, and $\frac{1}{2}$ is a fair estimate. On the other hand, $R_1 - R_{12}$ is a more accurate measure of the reward the rest of the system would get if A_2 chooses to do the subtask. As a result, by using $Rr_{12}^{(3)}$, k_2 would be reflecting A_2's attitude towards the rest of the system instead of its attitude towards A_1 alone. There are other ways to calculate the relational reward, but the three discussed here are among the most natural and intuitional.

The choice of relational reward calculation depends on the goal of the agents as well as the control mechanisms available to the system. In a cooperative system where the agents strive to maximize the system reward, one calculation of the relational reward is more expressive than another when it potentially allows a higher optimal expected social utility. We say that one calculation of the relational reward Rr_i is more *expressive* than another Rr_j if any relational reward that can be expressed using Rr_j can also be calculate with Rr_i. We have the following proposition about the expressiveness of the three relational reward calculations in a cooperative system. Its formal proof is based on the close form mathematical model we developed (Weber, Shen, & Lesser 2005).

Proposition 1. *In a cooperative system, when the reward splitting is fixed and attitude parameter k_2 can be varied between 0 and 1, $Rr_{12}^{(3)}$ is at least as expressive as $Rr_{12}^{(2)}$, and $Rr_{12}^{(2)}$ is at least as expressive as $Rr_{12}^{(1)}$. If the reward splitting is adjustable as well, then the expressiveness of $Rr_{12}^{(2)} \times Rr_{13}^{(2)}$ and $Rr_{12}^{(3)} \times Rr_{13}^{(3)}$ are the same, while $Rr_{12}^{(1)} \times Rr_{13}^{(1)}$ is less expressive.*

Optimality Graphs

In a cooperative system within a certain environment, there is a maximum expected utility for all settings of attitude parameters and reward splitting. It can be derived from the mathematical model we developed using OAR (Weber, Shen, & Lesser 2005). In this model, an agent's objective function is expressed in terms of the attitude parameter and reward splitting settings. Oftentimes, we want to show the optimality of a setting of the attitude parameters only, or a specific reward splitting alone. *Optimality graph* is designed

	r	dur	sl	e	R
T_1	30	[20,40]	[0,20]	[0,20]	30
T_{12}	30	[20,40]	[0,20]		
T_{13}	30	[20,40]	[0,20]		
T_2	30	[10,20]	[0,20]	[0,20]	[2,4]
T_3	50	[70,80]	[0,20]	[0,20]	[30,50]

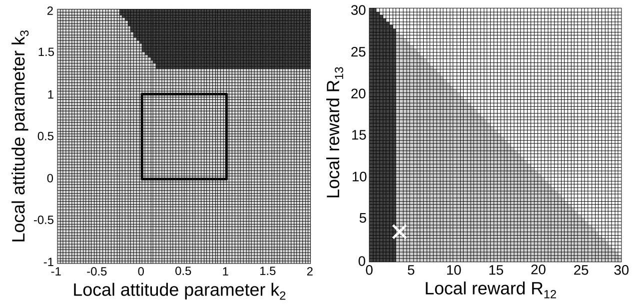
Table 1: Environmental parameters of Scenario 1

for this purpose. Its dimensions are labeled as the different attitude parameters or reward splittings. In order to examine the optimality of a certain setting of attitude parameters in the system, we fix the attitude parameters, vary the reward splitting and record the maximum social utility calculated using the mathematical model. If this maximum value is the same as the optimum achieved by varying all the different parameters, then we call this attitude parameter setting optimal. The corresponding point of this setting in the optimality graph is colored dark grey. If the setting is not optimal, the corresponding point is shown in light gray. When a setting of attitude parameters is chosen that lies within a dark gray area, it is possible to have a reward splitting such that the system achieves the optimal social utility in the current environment. For any reward splitting in a light gray area, the optimal social utility cannot be achieved no matter what the reward splitting is. Similarly, an optimality graph can be plotted for reward splitting as well to examine the optimality of different reward splitting settings. Figures 3(a) and 3(b) show examples of optimality graphs for a cooperative system. With the optimality graphs, we are able to discuss the interaction between different negotiation parameters without the mathematical details of the statistical model itself.

(Shen, Zhang, & Lesser 2004) demonstrates that attitude parameters can be used as an effective local mechanism to deal with the uncertainty associated with an external task that an agent receives from other agents. Unfortunately, there are cases where an attitude parameter $k \in [0, 1]$ cannot fully deal with such uncertainty and guarantee the optimal system performance. As an example, consider the scenario in Table 1 and the case where R_{12} is much bigger than R_2. In this situation, A_3 is unlikely to commit to T_{13}. Therefore there is very little chance that A_2 will actually get the R_{12} since the reward is awarded only if both T_{12} and T_{13} are completed. Thus A_2 is much better off ignoring the subtask and doing its local task. Unfortunately, even if we set k_2 to its lowest 0, A_2 will still choose to commit to T_{12} instead.

One way to deal with such uncertainty is to extend the range of the attitude parameters: $k_i \in [a^{k_i}, b^{k_i}]$, $a^{k_i} \le 0$, $b^{k_i} \ge 1$. As shown in Figure 3(a), this extended range can potentially lead to better system performance. This optimality graph is produced based on Scenario 1 with $k_i \in [-1, 2]$. The square in this graph denotes the original range of $k_i \in [0, 1]$. We can see that the optimal social utility can be achieved only with a $k_2 > 1.3$ and not the original range.

As we mentioned previously, reward splitting is very important for non-cooperative agents to improve their performance. In a cooperative system, the reward splitting may still improve the system performance, as shown in Figure 3(b). However, we proved that in a cooperative system, a



(a) (b)

Figure 3: (a) An attitude parameter optimality graph for Scenario 1 showing the need for larger attitude parameter ranges. (b) The reward splitting optimality graph for Scenario 1. The white cross denotes the setting from (Shen, Zhang, & Lesser 2004) that lies in the suboptimal area, marked in medium gray. This shows that different reward splittings can lead to higher gains for a cooperative system.

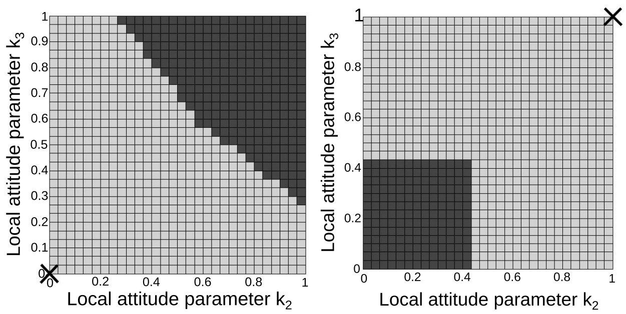
	r	dur	sl	e	R
T_1	40	[20,40]	[0,20]	[0,20]	30
T_{12}	40	[20,40]	[0,20]	-	-
T_{13}	40	[20,40]	[0,20]	-	-
T_2	50	[70,80]	[0,20]	[0,20]	[14,18]
T_3	50	[70,80]	[0,20]	[0,20]	[14,18]

Table 2: Environmental parameters of Scenario 2

reward splitting of assigning no real reward to any of the subtasks always lies in the optimal area, i.e., the optimal social utility can be achieved even when no real reward is given to the agents who complete the subtasks if all of the agents are cooperative (Weber, Shen, & Lesser 2005).We also discovered that the density of optimal solutions for a cooperative system is fairly high. In most environments a number of reward splittings may lead to the optimal behavior. This is particularly beneficial for a distributed system where the agent only has a local view of the system and can decide on the reward splitting with limited information. With the high density of optimal settings, the agent is more likely to choose an optimal reward splitting even with its limited local knowledge. We can also introduce a secondary goal to choose among the multiple optimal solutions. Examples for such goals include fairness and a minimal reward for the subtasks.

Using OAR

OAR is a formal framework that can be used to model different negotiation systems and study various negotiation strategies. (Shen, Zhang, & Lesser 2004) shows in detail how a statistical model is built for the example system shown in Figure 2 where the reward splitting is fixed. In their model a simple negotiation strategy was used. It shows us the relationship between the environment, the level of local cooperation and the global system performance in a formal clear way that allows us to explain system behavior and predict system performance. The analysis also results in a set of design equations that can be used directly to design distributed



(a) (b)

Figure 4: (a) An attitude parameter optimality graph, gathered from Scenario 2 for a cooperative object function. The black cross marks where $(k_2, k_3) = (0, 0)$, which is not an optimal solution. (b) An attitude parameter optimality graph, gathered from Scenario 3 for a cooperative object function. The black cross marks where $(k_2, k_3) = (1, 1)$, which is a suboptimal solution.

	r	dur	sl	e	R
T_1	40	[20,40]	[0,20]	[0,20]	30
T_{12}	40	[20,40]	[0,20]	-	-
T_{13}	40	[20,40]	[0,20]	-	-
T_2	40	[10,20]	[0,20]	[0,20]	[12,15]
T_3	40	[10,20]	[0,20]	[0,20]	[12,15]

Table 3: Environmental parameters of Scenario 3

local mechanisms that optimize the performance of the system dynamically. Similar models can be built for other systems and negotiation strategies (Zheng & Zhang 2005).

We distinguish the notion of "self-interested" versus "cooperative" from "self-directed" versus "externally-directed". The degree of cooperativeness illustrate the goal of an agent, while self-directness is the local mechanism used to achieve the goal. This is very different from traditional research, where the self-interested agents are assumed to be completely self-directed and cooperative agents are assumed to be completely externally-directed. In this section, we examine the traditional approaches and show that these two notions are not directly related to each other and the distinction needs to be made. We explore three different scenarios. Please refer to the appendix for the formulas used to generate the optimality graphs for these scenarios.

In the first scenario, the contractee agents are completely self-directed while the contractor agent is cooperative and wants to maximize the social utility of the system. As an example, in Figure 2 A_2 and A_3 are both completely self-directed, i.e., $k_2 = k_3 = 0$. A_1 is cooperative, and its objective function $O_1 = 1/2 \cdot \sum_i ER_i$, i.e., $w_1 = \frac{1}{2}$. Figure 4(a) shows the attitude parameter optimality graph for Scenario 2 from Table 2. The lower left corner, marked with a black cross, corresponds to $k_2 = k_3 = 0$ and is far from the optimal, dark gray area. This shows that in certain environments, no matter how much real reward incentive is given to the completely self-directed agents, A_1 cannot achieve its goal, i.e., the optimal social utility.

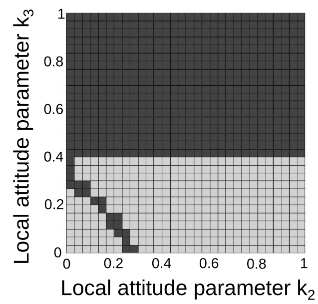


Figure 5: An attitude parameter optimality graph, gathered from Scenario 2 for a self-interested agent A_2 whose objective function is $O_2 = \sum ER_2$.

In the second scenario, let us assume that all the agents in the system are cooperative, but A_2 and A_3 are completely externally-directed. Table 3 is an example environment setting for the system shown in Figure 2 where $k_2 = k_3 = 1$ and $O_i = 1/2 \cdot \sum_i ER_i$. Figure 4(b) shows its attitude parameter optimality graph. The black cross in the graph, where $(k_2, k_3) = (1, 1)$, lies in the suboptimal area. This scenario demonstrates that even when the agents are completely externally-directed, the optimal social utility cannot be guaranteed no matter what the reward splitting is.

On the other hand, if an agent is self-interested and only tries to maximize its own reward, it may still be beneficial not to be completely self-directed all the time. We extend the definition of optimality graphs to represent non-cooperative objective functions. An attitude parameter (or reward splitting) setting is optimal if it can potentially maximize the specified objective function. Let us look at the example system with Scenario 2 where A_2 is self-interested, i.e., its objective function $O_2 = ER_2$. Figure 5 shows the attitude parameter optimality graph for $O_2 = ER_2$. As shown, in certain environments only when $k_2 \neq 0$ (i.e., A_2 is not completely self-directed) can it achieve the optimal local reward.

The counter-intuitive behaviors illustrated in the three scenarios are caused by the uncertainty related a task that requires multiple agents' cooperation. An agent often is not aware of other agents' agenda and cannot guarantee how their behaviors may affect its own return. For example, in an environment with high uncertainty, the system may gain more social utility if the agents choose ignoring the non-local tasks to avoid wasting resource unnecessarily. Similarly, a self-interested agent may occasionally decide to be externally directed towards a non-local task that has a good return and a high likelihood of being finished. Therefore, in order to best achieve its goal, an agent needs a local control mechanism such as the attitude parameter to respond to the uncertainties in the environment and make decisions that concern other agents. The scenarios discussed above demonstrate that the degree of cooperativeness and the degree of self-directedness are not directly related. It is necessary to make a clear distinction between them in order to optimize the agents' performance. This is the distinction between the goal of an agent and the local mechanism used to achieve the goal. The above examples also show that it is often the case that there is not one single attitude parameter setting that guarantees optimal performance in all environments. It is the same case for reward splitting. Therefore, it is beneficial for the agents to dynamically adjust their local parameters in response to the ever changing environment.

(Shen, Zhang, & Lesser 2004) discusses the dynamic adjustment of attitude parameters in a cooperative system. It is proven that it is safe for the agents to adjust their attitude parameters locally and reach a global equilibrium. In a non-cooperative system, i.e., when not all the agents in the system are cooperative, we need to take a game theoretic view and find a Pareto Nash Equilibrium for the different utility functions of the agents. This is part of our future research.

Conclusions and Future Work

In this paper, we introduced OAR, a formal framework to study different issues related to negotiation. It is designed to answer the question of how the agents should interact in an evolving environment in order to achieve their different goals. There are three components in OAR. Objective functions specify different goals of the agents involved in negotiation. Attitude parameters reflect the negotiation attitude of each agent towards another agent. Reward splitting specifies how a contractor agent divides the reward between the subtasks it needs to contract out. The traditional categorization of self-interested and cooperative agents is unified by adopting a utility view. Both attitude parameters and reward splitting can be used as effective local mechanisms for the agents to realize their goals. We illustrate empirically that the degree of cooperativeness of an agent and the degree of its self-directness are not directly related.

In our future work, we intend to use OAR to model and evaluate different systems with various negotiation strategies. Specifically, we are currently studying the role of decommitment penalty as a new parameter in OAR. In (Weber, Shen, & Lesser 2005) we start exploring the role of explicit decommitment in a more complex negotiation protocol using OAR framework with success. Another topic of interest is how the dynamic adjustment of local parameters may play out in a non-cooperative system. The study of such issues will help us to understand the behaviors of a system with complex agent interactions and guide us in the design process. In OAR, the objective function represents the goal of each agent from a local perspective. We are looking into explicitly representing other criteria that may be used to measure a system's overall performance such as fairness and load balancing.

Appendix

This appendix lists the formulas developed for the three agent model that are used to generate the optimality graphs in this paper. For their complete derivation, please refer to (Shen, Zhang, & Lesser 2004) and (Weber, Shen, & Lesser 2005).

Objective functions of the three agents:

$$\forall i, O_i = \sum_j ER_j.$$

When a task of type i arrives at a given time, the probability of there being a task of type j that has conflict with it:

$$\begin{aligned} & Pc_{ij} \\ = \ & P(sl_i - dur_j \leq est_j - est_i \leq dur_i - sl_j) \\ = \ & \sum_{z=-\infty}^{+\infty} \sum_{y=z}^{+\infty} \sum_{x=z}^{y} \\ & P_{est_j - est_i}(x) P_{dur_i - sl_j}(y) P_{sl_i - dur_j}(z), \end{aligned}$$

where

$$P_{est_j - est_i}(x) = \frac{1}{r_j},$$

$$\begin{aligned} & P_{dur_i - sl_j}(y) \\ = \ & \begin{cases} \frac{b_i^d - a_j^s - y}{(b_i^d - a_i^d)(b_j^s - a_j^s)}, & \\ \quad \max(a_i^d - a_j^s, b_i^d - b_j^s) < y < b_i^d - a_j^s; & \\ \frac{1}{b_i^d - a_i^d}, \quad a_i^d - a_j^s \leq y \leq b_i^d - b_j^s; & \\ \frac{1}{b_j^s - a_j^s}, \quad b_i^d - b_j^s \leq y \leq a_i^d - a_j^s; & \\ \frac{b_j^s + y - a_i^d}{(b_i^d - a_i^d)(b_j^s - a_j^s)}, & \\ \quad a_i^d - b_j^s < y < \min(a_i^d - a_j^s, b_i^d - b_j^s); & \\ 0, \quad otherwise. & \end{cases} \end{aligned}$$

and

$$\begin{aligned} & P_{sl_i - dur_j}(z) \\ = \ & \begin{cases} \frac{b_i^s - a_j^d - z}{(b_i^s - a_i^s)(b_j^d - a_j^d)}, & \\ \quad \max(a_i^s - a_j^d, b_i^s - b_j^d) < z < b_i^s - a_j^d; & \\ \frac{1}{b_i^s - a_i^s}, \quad a_i^s - a_j^d \leq z \leq b_i^s - b_j^d; & \\ \frac{1}{b_j^d - a_j^d}, \quad b_i^s - b_j^d \leq z \leq a_i^s - a_j^d; & \\ \frac{b_j^d + z - a_i^s}{(b_i^s - a_i^s)(b_j^d - a_j^d)}, & \\ \quad a_i^s - b_j^d < z < \min(a_i^s - a_j^d, b_i^s - b_j^d); & \\ 0, \quad otherwise. & \end{cases} \end{aligned}$$

The expected reward that A_2 or A_3 collects at each time unit:

$$ER_i = \frac{1}{r_i}(ER_i^{(1)} + ER_i^{(2)} + ER_i^{(3)} + ER_i^{(4)}) + \frac{1}{r_1} ER_i^{(5)},$$

where

$$ER_i^{(1)} = Pc_{1i,i} \cdot (1 - Pc_{ii}) \cdot E(R_i | R_i > Rn_i),$$

$$\begin{aligned} & ER_i^{(2)} \\ = \ & (1 - Pc_{1i,i}) \cdot Pc_{ii} \cdot [E(R_i | R_i > R_i') \\ & + \frac{1}{2} E(R_i | R_i = R_i')], \end{aligned}$$

$$\begin{aligned} & ER_i^{(3)} \\ = \ & Pc_{1i,i} \cdot Pc_{ii} \cdot [E(R_i | R_i > Rn_i \& R_i > R_i') \\ & + \frac{1}{2} E(R_i | R_i > R_{ni} \& R_i = R_i')], \end{aligned}$$

$$ER_i^{(4)} = (1 - Pc_{1i,i})(1 - Pc_{ii}) \cdot \frac{a_i^r + b_i^r}{2},$$

and

$$ER_i^{(5)} = R_{1i} \cdot Pcommit_2 \cdot Pcommit_3;$$

where

$$E(R_i|R_i > Rn_i) = \begin{cases} \frac{a_i^r+b_i^r+1}{2}, & \lfloor Rn_i \rfloor < a_i^r; \\ \frac{(b_i^r-\lfloor Rn_i \rfloor)(b_i^r+\lfloor Rn_i \rfloor+1)}{2(b_i^r-a_i^r)}, & a_i^r \leq \lfloor Rn_i \rfloor < b_i^r; \\ 0, & \lfloor Rn_i \rfloor \geq b_i^r. \end{cases}$$

$$E(R_i|R_i > R_i') = \sum_{y=a_i^r+1}^{b_i^r} \sum_{x=y+1}^{b_i^r} xP_{R_i}(x)P_{R_i}(y),$$

$$\frac{1}{2}E(R_i|R_i = R_i') = \frac{a_i^r + b_i^r + 1}{4(b_i^r - a_i^r)},$$

$$E(R_i|R_i > Rn_i \& R_i > R_i') = \frac{1}{(b_i^r - a_i^r)^2} \sum_{y=a_i^r+1}^{b_i^r} \sum_{x=\max(\lfloor Rn_i \rfloor+1, y+1)}^{br_i} x$$

$$\frac{1}{2}E(R_i|R_i > R_{ni} \& R_i = R_i') = \begin{cases} 0, & \lfloor Rn_i \rfloor \geq b_i^r; \\ \frac{(b_i^r-\lfloor Rn_i \rfloor)(b_i^r+\lfloor Rn_i \rfloor+1)}{4(b_i^r-a_i^r)^2}, & a_i^r \leq \lfloor Rn_i \rfloor < b_i^r; \\ \frac{a_i^r+b_i^r+1}{4(b_i^r-a_i^r)}, & \lfloor Rn_i \rfloor < a_i^r. \end{cases},$$

$$\begin{aligned} & Pcommit_i \\ = \; & Pc_{1i,i}(1 - Pc_{11})P(Rn_i \geq R_i) \\ & + \frac{1}{2}Pc_{1i,i} \cdot Pc_{11}P(Rn_i \geq R_i) \\ & + \frac{1}{2}(1 - Pc_{1i,i})Pc_{11} \\ & + (1 - Pc_{1i,i})(1 - Pc_{11}) \end{aligned}$$

and

$$P(Rn_i \geq R_i) = \sum_{x=a_i^r+1}^{\lfloor Rn_i \rfloor} P_{R_i}(x)$$

$$= \begin{cases} 1, & \lfloor Rn_i \rfloor \geq b_i^r \\ \frac{\lfloor Rn_i \rfloor - a_i^r}{b_i^r - a_i^r}, & a_i^r \leq \lfloor Rn_i \rfloor \leq b_i^r \\ 0, & \lfloor Rn_i \rfloor \leq a_i^r \end{cases}.$$

The expected reward that A_1 collects at each time unit:

$$ER_1 = \frac{1}{r_1} \cdot R_{11} \cdot Pcommit_2 \cdot Pcommit_3.$$

The optimal setting for k_2, k_3, R_{12} and R_{13} is:

$$\text{argmax}_{k_2,k_3,R_{12},R_{13}}(ER_1 + ER_2 + ER_3).$$

If we fix R_{12} and R_{13}, we can get the optimal attitude setting k_2 and k_3:

$$\text{argmax}_{k_2,k_3}(ER_1 + ER_2 + ER_3).$$

Similarly, if we fix k_2 and k_3, the optimal reward splitting is:

$$\text{argmax}_{R_{12},R_{13}}(ER_1 + ER_2 + ER_3).$$

References

Braynov, S., and Sandholm, T. 2002. Contracting with uncertain level of trust. *Computational Intelligence* 18(4):501–514.

Decker, K., and Lesser, V. 1993. An approach to analyzing the need for meta-level communication. *International Joint Conference on Artificial Intelligence* 1.

Jung, H.; Tambe, M.; and Kulkarni, S. 2001. Argumentation as distributed constraint satisfaction: Applications and results. In *Proceedings of the Fifth International Conference on Autonomous Agents*, 324–331. Montreal, Canada: ACM Press.

Klusch, M., and Gerber, A. 2002. Dynamic coalition formation among rational agents. *IEEE Intelligent Systems* 17(3):42–47.

Levine, D. K. 1998. Modeling altruism and spitefulness in experiments. *Review of Economic Dynamics* 1:593–622.

Saha, S.; Sen, S.; and Dutta, P. S. 2003. Helping based on future expectations. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, 289–296. Melbourne, Australia: ACM Press.

Sandholm, T.; Sikka, S.; and Norden, S. 1999. Algorithms for optimizing leveled commitment contracts. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 535–540.

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

Sen, S. 2002. Believing others: Pros and cons. *Artificial Intelligence* 142(2):179–203.

Shen, J.; Zhang, X.; and Lesser, V. 2004. Degree of local cooperation and its implication on global utility. In *Proceedings of Third International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2004)*, volume 2, 546–553. New York, New York: IEEE Computer Society.

Vidal, J. M. 2004. The effects of cooperation on multiagent search in task-oriented domains. *Journal of Experimental and Theoretical Artificial Intelligence* 16(1):5–18.

Weber, I.; Shen, J.; and Lesser, V. 2005. Modeling and analyzing cooperation parameters in a multi-agent system. Technical Report 05-29, Computer Science Department, University of Massachusetts, Amherst.

Zhang, X.; Lesser, V.; and Wagner, T. 2003. Integrative negotiation in complex organizational agent systems. In *Proceedings of the 2003 IEEE/WIC International Conference on Intelligent Agent Technology (IAT 2003)*, 140–146.

Zheng, Q., and Zhang, X. S. 2005. The study of virtual organizations using multi-agent system. In *Proceedings of Fourth International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2005), to appear*.