# A Formal Study of Distributed Resource Allocation Strategies in Multi-Agent Systems

Jiaying Shen, Micah Adler, Victor Lesser

Department of Computer Science University of Massachusetts Amherst, MA 01003

### Abstract

In multi-agent systems, centralized optimal solutions are often impractical due to scalability and resource limitation issues which makes simpler distributed algorithms preferable. Unfortunately, there has been little work that formally studies different distributed systems to predict their performance or explain their behavior. In this work, we study three different distributed resource allocation strategies in a simple problem setting. We built a formal model that predicts the performance of different systems and verified the result through simulation. The performance of the distributed algorithms are compared to the centralized optimal solution. This work shows that it is possible to build a formal model for a distributed resource allocation problem. The simulation results shed some light on the advantages and disadvantages of a centralized solution and various distributed solutions.

### 1 Introduction

In multi-agent systems, centralized optimal solutions are often impractical due to scalability and resource limitation issues which makes simpler distributed algorithms preferable. Unfortunately, there has been little work that formally studies different distributed systems to predict their performance or explain their behavior. This is mainly due to the complexity of such systems. The only papers we are aware of that formally analyze a multi-agent system are [Decker and Lesser, 1993] and [Sen and Durfee, 1998].

In multi-agent systems, the distribution of resources among agents frequently does not match the individual needs of each agent. Hence, designing an algorithm to assign resources to agents to maximize the social utility is an important research topic. While a centralized solution can be designed to find the optimal allocation, an agent has to be selected to collect all the necessary information from the numerous other agents and designate the final allocation. As a result, there is inevitably a bottleneck in the system and the amount of communication needed is not trivial. More importantly, in many cases, computing an optimal solution is simply intractable. This all adds to the attraction of a distributed system whose interaction is strictly local. If the resource allocation decision is distributed to the local agents which only make local interactions, only local information is necessary, and the decision process can be much simpler compared to a centralized system. Unfortunately, since an agent makes its decision based only on local information, the bounded rationality limits the quality of the overall solution and the performance of different distributed algorithms is often ad-hoc and hard to predict. Even though there is much work in the sciences of complexity studying the formation of emergent behaviors from simple local interactions [Epstein and Axtell, 1996], few formalization results have been produced to explain such behavior. We are trying to model the distributed resource allocation problem in a formal way and thus be able to predict and compare the performance of different systems.

In this paper, we study three simple distributed resource allocation strategies in a simple problem setting. A formal model was built to predict the performance of the different strategies. The simulation results confirm the formal prediction. The performance of the distributed systems is also compared to the centralized optimal solution to see how good a result the systems with only local interactions generate. This work shows that it is possible to build a formal model for a distributed resource allocation problem. The simulation results also shed some light on the advantages and disadvantages of a centralized solution and different distributed solutions.

# 2 Problem Settings

- 1. We have a large collection of agents  $\{A_i | 1 \le i \le n\}$ . We also assume that the agents form a ring, such that every agent  $A_i$  has two neighbors  $A_{i-1}$  and  $A_{i+1}$ .
- 2. The order of events at each time step is as follows: resource regeneration, resource exchange if any, task execution if possible.
- 3. The resource is regenerated at every agent  $A_i$  at *Regeneration rate*  $R_i$  at each time step.  $R_i$  is uniformly distributed among agents over [a, b).

$$P_{R_i}(r) = \begin{cases} 1/(b-a), & a \le r < b; \\ 0, & \text{else.} \end{cases}$$
(1)

4. We define the resource at time step t for agent  $A_i$  before resource exchange but after resource regeneration as *pre-wealth*, and that after resource exchange and task execution as *post-wealth*. We denote them as  $W_{i,t}$  and  $W'_{i,t}$  respectively. There is no limit to the amount of resource an agent can accumulate, i.e., we have  $W_{i,t} \in [0, +\infty)$ and  $W'_{i,t} \in [0, +\infty)$ .

5. There is only one type of task done by the agents which generates equal utility. It takes one time unit to finish and an agent consumes *c* units of resource to execute it. An agent accumulates *utility* as it completes its task.

$$U_{i,t} = \begin{cases} U_{i,t-1} + 1, & \text{if } A_i \text{ finishes its task at time } t; \\ U_{i,t}, & \text{otherwise.} \end{cases}$$
(2)

- 6. *Limited rationality*: Each agent only has knowledge of its neighbors and can only exchange resources with its neighbors. This leads directly to strict locality.
- 7. Each time step each agent can initiate a resource exchange with at most one of its neighbors. We denote the amount of resource that  $A_i$  gets from its neighbor as  $e_{i,t}$ .  $e_{i,t}$  is decided by the resource allocation strategy employed by the agent. Obviously,  $e_{i,t} = 0$  if there is no resource exchange for  $A_i$  at time t.
- 8. Every agent  $A_i$  completes its current task at time t if and only if the sum of its pre-wealth  $W_{i,t}$  and the amount of resource it gets from its neighbor  $e_{i,t}$  is greater than or equal to the task's consumption rate, i.e.,  $W_{i,t} + e_{i,t} \ge c$ . Otherwise the current time unit is wasted. Hence, (2) is equivalent to:

$$U_{i,t} = \begin{cases} U_{i,t-1} + 1, & \text{if } W_{i,t} + e_{i,t} \ge c; \\ U_{i,t}, & \text{otherwise.} \end{cases}$$
(3)

9. We have the recursive formula for the agents' wealth:

$$W_{i,0} = R_i,$$
  
 $W_{i,t} = W'_{i,t-1} + R_i$ 
(4)

$$W'_{i,t} = \begin{cases} W_{i,t} + e_{i,t} - c, & \text{if } W_{i,t} + e \leq c; \\ W_{i,t} + e_{i,t}, & \text{otherwise.} \end{cases}$$
(5)

- 10. All the agents are sincere in the sense that the information they provide to their neighbors is correct.
- 11. There is no communication cost or resource exchange delay. The tasks are independent of each other.

There are two main reasons why we are trying to make our agents very simple. One of them is that a simple model will make it easier to formally study the characteristics of the different distributed resource allocation strategies. Another more important reason is that the beauty and advantage of a distributed solution lies in its simplicity. The main goal of this paper is to study the characteristics and performance of distributed agents with limited local interaction. This simple model will be extended in our future research, which is discussed in more detail in Section 7.

# **3** Distributed Resource Allocation Strategies

In a centralized resource allocation strategy, an agent is usually chosen to collect the information (current wealth and consumption rate) from all of the agents in the system and decide how to allocate the resource efficiently to complete the most number of tasks, which yields the most social utility. In the distributed strategies we are looking at here, the information an agent can get is strictly local. In other words, an agent only knows of the resource level of its neighbors, and can only adjust resource allocation through exchanging resource locally. Each such strategy has three components: when to exchange resource, with whom to exchange, and how much to exchange. We will take a look at them one by one.

#### 3.1 When to Exchange Resource

In most environments an agent does not want to be involved in unnecessary communication and resource exchange. As a result, it would not be a good idea to exchange resources every time. Hence, a simple and reasonable strategy to decide when to exchange resource would be:

When to exchange resource policy (PT): An agent initiates resource exchange negotiation when and only when it cannot execute its task at this time unit. Written formally,  $A_i$ initiates resource exchange negotiation with  $A_{i-1}$  or  $A_{i+1}$  if and only if  $W_{i,t} < c$  at time t.

This is a rather myopic policy, as the agent is only concerned about the current time step. But for this project, we will fix the "when" part of our strategies as PT. Later on, we can extend this policy to farther sighted ones, i.e., where agents will look n steps ahead.

#### **3.2** With Whom to Exchange Resource

We assume that each agent can exchange resource with at most one of its neighbors, and as a result, we need a policy to decide with whom to do the transaction. Although the agents in our system are inherently cooperative, they are somewhat self-interested in the sense that they want to gather as much wealth as they can in order to complete more tasks. After all, the social utility is the sum of the local utilities. Since an agent cannot have a global view of the entire system and allocate resource accordingly, a most reasonable strategy would be to exchange resource with the neighbor that has more extra resource. The heuristic behind this policy is that the more wealth an agent can gather, the more tasks it will be able to complete. Furthermore, the history can often be used as a good prediction for the future. The neighbor with greater extra resource now is more likely to have greater extra resource in the future and therefore have less problem finishing its own task later. Hence, we have the following policy:

With whom to exchange resource (PW): The neighbor with the greatest extra resource. If we denote the agent who is acquiring the resource as B (Borrower), its neighbors N(B), and the agent who is providing resource as L (Loaner), then the rule can be mathematically formulated as:  $L = argmax_{A_i \in N(B)}W_{i,t} - c$  and W(L) > c.

### 3.3 How Much Resource to Exchange

After deciding when and with whom to exchange resource, the next step and the most important step is to decide how much resource to exchange with the provider. This is the most interesting part of the strategy. Since an agent only has local view of itself and its resource provider, deciding how to distribute resource between the two of them is essentially doing its part deciding the global resource allocation. As we said before, in our strategies, an agent will initiate a resource exchange negotiation if and only if it does not have enough resource to complete the current task, and its neighbor will provide resource if and only if it has extra resource after completing the current resource. Hence, for borrower  $A_i$  and loaner  $A_j$ , we have  $0 \le e_{i,t} \le W_{j,t} - c$ .

Now the question is when the loaner  $A_j$  does not have enough extra resource for the borrower  $A_i$  to complete its current task, whether it should still give resource to the borrower and how much it should be. Written out explicitly, if  $W_{j,t} - c < c - W_{i,t}$ , what should  $e_{i,t}$  be? Similarly, if  $A_j$  has more than enough extra resource for  $A_i$  to complete its current task, how much should they exchange? In other words, when  $W_{j,t} - c > c - W_{i,t}$ , what should  $e_{i,t}$  be?

To answer these two questions, we have developed the following three different strategies:

**Mediocre Strategy:** The loaner will give as much resource as it can to help the borrower out but no more than needed.

$$e_{i,t} = \begin{cases} W_{j,t} - c, & \text{if } 0 \le W_{j,t} - c < c - W_{i,t}; \\ c - W_{i,t}, & \text{if } 0 \le c - W_{i,t} \le W_{j,t} - c; \\ 0, & \text{if } W_{i,t} > c \text{ or } W_{j,t} < c. \end{cases}$$
(6)

In this strategy, the loaner is not too selfish since it gives out as much resource as it can even when it cannot really help out the borrower. Neither is it too selfless since it will give at most what the borrower needs for the current step. Hence, we call this strategy *mediocre strategy*.

**Selfish Strategy:** The loaner will only give the exact amount of resource that the borrower needs to complete its current task if it has enough extra. Otherwise, it will give none.

$$e_{i,t} = \begin{cases} c - W_{i,t}, & \text{if } 0 \le c - W_{i,t} \le W_{j,t} - c; \\ 0, & \text{otherwise.} \end{cases}$$
(7)

We call this strategy *selfish strategy* as the loaner does not want to give resource even if it has extra as long as it does not have enough for the borrower to finish the current task.

**Selfless Strategy:** The loaner will give as much resource as it has extra, even if it is more than the borrower needs for the current time step.

$$e_{i,t} = \begin{cases} W_{j,t} - c, & \text{if } 0 \le W_{j,t} - c; \\ 0, & \text{otherwise.} \end{cases}$$
(8)

This is the most selfless strategy we are studying in the sense that the loaner is helping out the borrower as much as it can without considering the next time step for itself, and therefore we call it *selfless strategy*.

# 4 A formal study

In this section we will use statistical methods to analyze the three different protocols presented in the last section and try to predict how good they will perform compared to the centralized optimal solution. The reason why we chose statistical methods is based on the nature of our problem setting. There is a large collection of agents and the resource distribution is inherently statistical. In our analysis, we will first derive the probabilistic distribution of the wealth of an agent and thereafter its utility distribution at each time step. The expected values can then be calculated and compared to those of a centralized solution. The full derivation will be detailed in a technical report.

#### 4.1 Centralized Optimal Solution

In order to see how well a distributed solution can perform, we will need to analyze the optimal solution first to serve as a base line for comparison.

In this paper, we will ignore the details of the centralized algorithm. All we are concerned with is that the centralized solution makes use of as much social wealth as possible and generates as much social utility as possible. If we denote the social utility at time t as  $U_t$ , the social pre-wealth as  $W_t$  and the social post-wealth as  $W'_t$ , we have the following result:

$$U_t = \min(|W_t/c|, n) \tag{9}$$

$$W_0 = E(R) * n \tag{10}$$

$$W_t = W'_{t-1} + n * E(R) \tag{11}$$

$$W'_{t} = W_{t} - U_{t} * c. (12)$$

E(R) is the expected regeneration rate. Since the regeneration rate is uniformly distributed, E(R) = (a + b)/2.

### 4.2 Without Resource Exchange

The performance of a centralized optimal solution is the upper bound of a distributed solution, while the performance of a system in which there is no resource exchange or allocation at all is the lower bound. In such a system, each agent is isolated from others and can use its own resource as best as it can without reallocating at all. The analysis of such a system provides us with another baseline for comparison. For such a system, we have the following result: For agent  $A_i$ ,

$$W_{i,0} = R_i \tag{13}$$

$$W_{i,t} = W'_{i,t-1} + R_i$$
 (14)

$$W'_{i,t} = \begin{cases} W_{i,t}, & \text{if } W_{i,t} < c \\ W_{i,t} - c, & \text{otherwise.} \end{cases}$$
(15)

$$U_{i,t} = \begin{cases} U_{i,t-1} + 1, & \text{if } W_{i,t} \ge c \\ U_{i,t-1}, & \text{otherwise.} \end{cases}$$
(16)

We can further calculate the distribution of the post-wealth of  $A_i$  at time t:

$$P_{W'_{i,t}}(w) = \begin{cases} \frac{t - \max(0, \lceil \frac{(t+1)a - w}{c} \rceil) + 2}{b - a}, & 0 \le w < (t+1)(b-c) \\ \frac{t - \max(0, \lceil \frac{(t+1)a - w}{c} \rceil) + 1}{b - a}, & (t+1)(b-c) \le w < c \\ \frac{1}{b - a}, & c \le w < (n+1)(b-c) \\ 0, & \text{otherwise.} \end{cases}$$

The recursive formulas for  $W_{i,0}$ ,  $W_{i,t}$  and  $U_{i,t}$  will stay the same for the three different strategies. What differs is  $W'_{i,t}$ . We will derive them one by one.

### 4.3 Mediocre Strategy

In the Mediocre Strategy, the resource exchange will happen only when  $W_{i,t} < c$  and  $W_{j,t} \ge c$ . If  $W_{j,t} - c < c - W_{i,t}$ , the loaner's extra resource is still not enough for the borrower to complete its task,  $A_i$ 's post-wealth will be  $W_{i,t} + W_{j,t} - c$ . If this is not the case, the loaner will give the borrower just enough resource to complete its current task, and  $A_i$ 's postwealth will be 0. Thus, when  $W_{i,t} < c$  and  $W_{j,t} \ge c$ , we have the following:

$$W'_{i,t} = \begin{cases} W_{i,t} + W_{j,t} - c, & \text{if } W_{i,t} + W_{j,t} < 2c \\ 0, & \text{if } W_{i,t} + W_{j,t} \ge 2c \end{cases}$$
(17)

When there is no resource exchange happening, we have:

$$W'_{i,t} = \begin{cases} W_{i,t}, & \text{if } W_{i,t} < c \text{ and } W_{j,t} < c \\ W_{i,t} - c, & \text{if } W_{i,t} \ge c \end{cases}$$
(18)

The recursive expression for the distribution of  $W_{i,t}$  is:

$$P_{W_{i,t}}(w_{i,t}') = \begin{cases} \sum_{w=0}^{c-1} (P_{W_{i,t}}(w) \sum_{m=2c-w}^{\infty} P_{W_{i,t}}(m)) \\ +P_{W_{i,t}}(c), & \text{if } w_{i,t}' = 0 \\ \sum_{w=0}^{w'i,t-1} (P_{W_{i,t}}(w) P_{W_{i,t}}(c+w_{i,t}'-w)) \\ +P_{W_{i,t}}(w_{i,t}'+c) \\ +P_{W_{i,t}}(w_{i,t}') \sum_{w=0}^{c} P_{W_{i,t}}(w), \\ P_{W_{i,t}}(w_{i,t}'+c), & \text{if } 0 < w_{i,t}' < c \end{cases}$$

### 4.4 Selfish Strategy

In the Selfish Strategy, there is a chance of the resource exchange happening when  $W_{i,t} < c$  and  $W_{j,t} \geq c$ . If  $W_{j,t} - c \geq c - W_{i,t}$ , the loaner will give just enough resource to help the borrower to complete its task and the borrower's post-wealth will be 0, otherwise, the loaner will not give resource to the borrower even if it has extra. Thus, when  $W_{i,t} < c$  and  $W_{j,t} \geq c$ , we have:

$$W'_{i,t} = \begin{cases} 0, & \text{if } W_{i,t} + W_{j,t} \ge 2c \\ W_{i,t}, & \text{if } W_{i,t} + W_{j,t} < 2c \end{cases}$$
(19)

When there is no resource exchange happening, we still have (18). The recursive expression for the distribution of  $W_{i,t}$  is :

$$P_{W'_{i,t}}(w'_{i,t}) = \begin{cases} \sum_{w=0}^{c-1} (P_{W_{i,t}}(w) \sum_{m=2c-w}^{\infty} P_{W_{i,t}}(m)) \\ +P_{W_{i,t}}(c), & \text{if } w'_{i,t} = 0 \\ P_{W_{i,t}}(w'_{i,t}) \sum_{w=0}^{2c-w'_{i,t}-1} P_{W_{i,t}}(w) \\ +P_{W_{i,t}}(w'_{i,t}+c), & \text{if } 0 < w'_{i,t} < c \\ P_{W_{i,t}}(w'_{i,t}+c), & \text{if } w'_{i,t} \ge c \end{cases}$$

### 4.5 Selfless Strategy

In the Selfless Strategy, the resource exchange will happen when  $W_{i,t} < c$  and  $W_{j,t} \ge c$ . If  $W_{j,t} - c < c - W_{i,t}$ , the loaner's extra resource is still not enough for the borrower to complete its task,  $A_i$ 's post-wealth will be  $W_{i,t} + W_{j,t} - c$ . If this is not the case, the borrower can execute the task and its post-wealth will be  $W_{i,t} + W_{j,t} - 2c$ . Thus, when  $W_{i,t} < c$ and  $W_{j,t} \ge c$ , we have the following:

$$W'_{i,t} = \begin{cases} W_{i,t} + W_{j,t} - c, & \text{if } W_{i,t} + W_{j,t} < 2c \\ W_{i,t} + W_{j,t} - 2c, & \text{if } W_{i,t} + W_{j,t} \ge 2c \end{cases}$$
(20)

When there is no resource exchange happening, we still have (18). The recursive expression for the distribution of  $W_{i,t}$  is:

$$= \begin{cases} P_{W'_{i,t}}(w'_{i,t}) \\ \sum_{w=0}^{w'_{i,t}} P_{W_{i,t}}(w)[P_{W_{i,t}}(w'_{i,t} - w + c) \\ +P_{W_{i,t}}(w'_{i,t} - w + 2c)] \\ +P_{W_{i,t}}(w'_{i,t})\sum_{w=0}^{c-1} P_{W_{i,t}}(w) \\ +P_{W_{i,t}}(w'_{i,t} + c), & \text{if } 0 \le w'_{i,t} < c \\ P_{W_{i,t}}(w'_{i,t} + c), & \text{if } w'_{i,t} \ge c. \end{cases}$$

# 5 Simulation Results





(b) Social wealth of different systems.

Figure 1: Social utility and social wealth of a group of 1000 agents. a = 10, b = 45, c = 30. The selfless agents perform the best among the distributed systems. The selfish and mediocre strategies perform about the same.

We have built a simulator to simulate the environment with the problem settings described in Section 2. In every simulation run we have five groups of agents. Each group is comprised of 1000 agents, each of which have two neighbors. Every agent in a group has the same regeneration rate and consumption rate as their counterpart in the other four groups. The only difference between the five corresponding agents is their strategies for resource exchange. The first one does not do any resource change, the second one is using the centralized strategy, while the other three are using the three distributed strategies as defined in Section 3. If we plot out the expected post-wealth accumulation of the different systems based on the distribution functions generated in Section 4 and the expected social utility value, we get graphs similar to Figure 1. This predicts that among the different distributed resource allocation strategies, the selfless strategy will generate the most social utility.

Figure 1 shows the comparison of the total wealth and social utility over time from our simulation. In this simulation, we set the lower bound of the agent regeneration rate to 10, upper bound to 45, and the consumption rate to 30. From Figure 1(a), we can see that the centralized solution indisputably performs the best. All three distributed solutions perform considerably better than the lower bound agents, i.e., those without any resource exchange. Figure 1(b) shows the social wealth over time of the five groups of agents. While the centralized solution makes the best usage of the social wealth, the non-exchanging resource version wastes a lot. The three distributed versions are somewhat in between these extremes. We can also see that although the performance of the three distributed versions is very close to each other, the Selfless Strategy uniformly performs better than the other two. This simulation result conforms with the prediction of the formal analysis.

Now let us fix a and b, and vary the consumption rate. We also divide the social utility of the three distributed groups and the non-exchanging resource group by that of the centralized solution to see how well they perform against each other. Figure 2(a) shows again that the distributed strategies has a clear advantage, and the selfless strategy performs the best. What is interesting is the shape of the curves. When the consumption rate draws near to both ends of the regeneration rate range, all five groups perform closely to each other, while when the consumption rate is close to the average regeneration rate, the centralized group performs considerably better than the three distributed groups. This phenomena corresponds well with our intuition. When the consumption rate is high, the resource is so tight that no agent is particularly abundant in wealth, and the global view has little use. When the consumption rate is low, most agents can be self-satisfying, a global view does not have clear advantage either. Only when the consumption rate is in the middle range will the variance among the agents make a difference and only then will resource exchange become important for utilizing the social wealth and achieving more social utility.

**Definition 1** When a group of agents' regeneration rates are uniformly distributed between a and b, we define the **abundance rating** v of the agents' regeneration rates with regard to the consumption rate c as:

$$v = \frac{\sum_{i=1}^{n} (R_i - c)^2}{n - 1}$$

This is an indication of how abundant the resource of a system is compared to its consumption rate. When the resource is abundant or too scarce, the rating is low. We use it to see the relationship of the distribution of the regeneration rates and the consumption rate. Figure 2(b) together with 2(a) shows that the smaller the abundance rating is, the more difference it makes to choose a better strategy.



(a) The performance of a group of 1000 agents when the consumption rate varies. a = 10, b = 45.



(b) The variance of the regeneration rate with regard to the consumption rate. a = 10, b = 45.

Figure 2: The performance of the different distributed strategies as compared to the variance of the regeneration rate with regard to the consumption rate. a = 10, b = 45.

# 6 Adding in Resource Exchange Cost

We have so far only considered the evaluation of social utility collected by the agents. In this section, we will compare the resource exchange cost of different systems in order to achieve the social utility. We define the cost of a resource exchange between a pair of agents as the product of the quantity of resource exchanged and the distance between the agents.

Whereas the calculation of the resource exchange cost of the distributed systems is trivial, it is not easy to find the optimal resource allocation policy for a centralized system that achieves the optimal social utility. We designed an algorithm based on Minimum Cost Maximum Flow algorithms [Rosen, 2000] that generates the optimal resource allocation strategy for the centralized system in pseudo-polynomial time. It is optimal in the sense that it has the least resource exchange cost while still achieving the optimal social utility.



Figure 3: The resource exchange cost of different systems. a = 10, b = 45, c = 30



Figure 4: The ratio of social utility and resource exchange cost of different systems. a = 10, b = 45, c = 30

As seen in Figure 3, in order to achieve the optimal social utility, the centralized system suffers from the highest resource allocation cost. This is an interesting result, but not counter-intuitive. The marginal gain for the cost of more resource allocation starts to decrease significantly after certain a level of social utility is achieved. In order to achieve the optimal social utility, the centralized system has to pay a high price in the form of resource allocation cost. On the other hand, the Selfish strategy has the least resource allocation cost among the distributed versions. What we deem more interesting is the comparison of the utility/cost ratio of the different distributed resource allocation strategies, as shown in Figure 4. Again, the centralized system has the least utility/cost ratio. Among the distributed systems, the more selfish the resource exchange strategy is, the higher the utility/cost ratio is.

Our results indicate that when the only concern of a distributed resource allocation system is the social utility and not the resource allocation cost, the selfless strategy performs the best. Nevertheless, if lower resource allocation cost is important, then the more selfish system will give a better return on the social utility gained per resource allocation cost. This is an interesting observation. We plan to add a parameter to the strategies that changes the degree of self-interestedness to further study its relation with the performance.

It is unfair to conclude that the centralized system will give a lower utility/cost ratio than a distributed system. If we assign a different social utility function which takes the resource allocation cost into account, the optimal resource allocation found by the centralized system will no doubt outperform the distributed systems. The main disadvantage of the centralized system lies in its more complex system and algorithm design. With a more complex utility function, searching for an optimal solution might not be practical.

# 7 Conclusions and Future Directions

We used statistical methods to build a formal model of the distributed resource allocation problem of a simple setting. The simulation results conform well with the prediction of the model. Even with this simple problem setting, the model we built is already quite complex. This demonstrates the complexity of multi-agent systems. Nevertheless, our work shows that it is possible to formally study the characteristics and performance of a distributed resource allocation system. The statistical techniques are suitable for the analysis of a large distributed system and can be extended to more complex settings.

In the current model the agents form a ring. This is one of the simplest neighborhood designs, and thus puts our distributed resource exchanging policies on the simple end. Our next step will be to increase the number of neighbors an agent has and see how the performance of the distributed versions will change. By doing this, we essentially increase the individual agents' local view, enabling us to see the implication of the scope of an agent's local view on its performance.

We can also increase the intelligence of the agents without losing the simplicity of the distributed versions. The current model can be extended to allow local information to propagate to remote agents by adding a short term memory to each agent. This indirectly increases the local view, which we can compare to the more direct method mentioned above. By comparing systems with a different scope of local view and memory, we can study the relationship between the importance of a more global view and the resource constraints in the system.

# References

- [Decker and Lesser, 1993] Keith S. Decker and Victor R. Lesser. An approach to analyzing the need for meta-level communication. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, Chambry, August 1993.
- [Epstein and Axtell, 1996] Joshua M. M. Epstein and Robert L. Axtell. *Growing Artificial Societies: Social Science from the Bottom Up.* MIT Press, 1996.
- [Rosen, 2000] Kenneth H. Rosen, editor. Handbook of Discrete and Combinatorial Mathematics. CRC Press, 2000.
- [Sen and Durfee, 1998] Sandip Sen and Edmund H. Durfee. A formal study of distributed meeting scheduling. *Group Decision and Negotiation*, 7:265–289, 1998.