Combining Bundle Search with Buyer Coalition Formation in Electronic Markets: A Distributed Approach through Explicit Negotiation

Linli He
Department of Computer Science,
Texas A&M University,
College Station, TX77843-3112, USA
linli@cs.tamu.edu

Thomas R. Ioerger
Department of Computer Science,
Texas A&M University,
College Station, TX77843-3112, USA
ioerger@cs.tamu.edu

Abstract

In electronic markets, both bundle search and buyer coalition formation are profitable purchasing strategies for buyers who need to buy small amount of goods and have no bargaining power. It is valuable to combine these two purchasing strategies for buyers to obtain greater discounts based on the different discount policies of multiple sellers. In this paper, we present a distributed mechanism that allows buyers to use both purchasing strategies. The mechanism includes a very efficient heuristic bundle search algorithm and a distributed coalition formation scheme, which is based on an explicit negotiation protocol. The resulting coalitions are stable in the core. The simulation results show that the cost to buyers is close to the optimal cost. Increasing the number of buyers who are involved in the coalition formation process does not increase the communication load caused by the negotiation between two buyers.

Keywords

Bundle search, coalition formation, automated negotiation.

1. Introduction

In electronic markets, the distance among producers, wholesalers, distributors, retailers, and consumers has disappeared to approach zero [1]. There are many more choices faced by all parties involved in an electronic combinatorial trade than in a traditional trade. The relationship between suppliers and customers is going to be revolutionarily changed.

Buyers vary a great deal in the quantity of goods they purchase, in customer service requirements, in income, in time constraints and in many other dimensions. Different purchasing goals can cause widely varying production and transaction costs. Suppliers have their own "buyer selection" strategies to make better profitability [5]. Quickly differentiating the supplier's marketing strategy based on the difference of purchasing goals among various buyers plays a key role in improving the sellers' competitive capabilities in electronic markets.

On the other hand, buyers can build corresponding purchasing strategies to minimize their cost. In the traditional markets, it was impractical for buyers to build such purchasing strategies because of expensive product information access cost. However, in the age of electronic commerce, buyers can access product information easily and inexpensively. For suppliers, bundling large numbers of goods can be surprisingly profitable [4]. Conversely, buyers can build a corresponding bundled purchasing strategy to obtain greater discounts.

A well-known example of building such a purchasing strategy for buyers is to form buyer coalitions (Buyer Club) to enlarge the total quantity of goods purchased in each transaction. Buyers can obtain lower prices without buying more than their real demand [1,7,8,9]. If the buyers are heterogeneous in the sense that they need to buy different goods in a combinatorial market, the buyer coalition formation is the so-called combinatorial coalition formation [8]. Most previous work is based on an assumption, under which the price of goods is a function of the number of items sold in each transaction. The problem is to find the optimal coalition structure [12,13] that causes the lowest prices for buyers.

Another very interesting buyer strategy is called the "bundle search", which addresses the situation where a buyer needs to buy different goods as a bundle. A typical example is the travel package search problem [6]. Because of the different retail prices and discount policies of different suppliers, different bundles result in different discounts. The problem is to find the optimal bundle that results in minimum cost. Since the goods in a bundle can be different items, the discount policy is based on the total cost of all goods in each transaction.

Actually, we can view searching the maximal discount of a buyer coalition as a bundle search problem if the discount policies of sellers are based on the total cost to all buyers in the buyer coalition. Under this kind of discount policy, it is valuable for buyers to use both bundle search and buyer coalition formation to obtain better discounts.

There is little research work that considers both combinatorial coalition formation and bundle search together when the discount policies of sellers depend on the total cost of all goods sold in each transaction. One of the reasons can be that searching for the optimal buyer coalition structure and finding the optimal bundle are both computationally intractable problems. Finding the optimal buyer coalition structure can be translated into the weighted set packing problem [7], which is a NP-complete problem. Finding the optimal bundle is a NP-hard knapsack problem [6].

In this paper, we consider a purchasing problem in which a group of buyers are shopping from a certain group of sellers in a combinatorial market. The buyers have different shopping lists, and they are self-interested and geographically distributed. Different sellers offer different retail prices and different discount policies based on the total cost of all goods sold in one transaction. The problem is to find the optimal purchasing strategies that minimize the cost to buyers. Our approach is combining bundle search and buyer coalition formation through a distributed mechanism.

The rest of this paper is organized as follows: Section 2 gives a formal definition of the combinatorial purchasing problem we want to address in this paper. Section 3 describes a traditional centralized approach to solving this problem. The computational cost of the algorithm prohibits applying it to real applications. Section 4 describes our distributed approach to solving this problem, which includes an efficient heuristic algorithm for the bundle search problem and a distributed coalition formation scheme through an explicit negotiation protocol. Section 5 presents the simulation results. Finally, Section 6 gives the conclusions and discussions on future work.

2. Problem Formalization

2.1. Formal Problem Definition

Let $G = \{g_0, g_1, \ldots, g_{l-1}\}$ denote the collection of goods items. There is a group of buyers $B = \{b_0, b_1, \ldots, b_{m-1}\}$, each of them has a shopping list denoted by vector $Q_i = (q_{i0}, q_{i1}, \ldots, q_{i,l-1})$, where q_{ik} refers to the quantity of each item g_k buyer b_i needs to buy ($i = 0, 1, .., m-1; k = 0, 1, .., l-1$). There is a set of sellers $S = \{s_0, s_1, \ldots, s_{n-1}\}$ who can supply partial or all goods in G. Each seller s_j ($j = 0, 1, .., n-1$) has its own discount function $\delta_j(c)$: $R^+ \rightarrow R^+$, which is the discount a buyer obtains when the cost of his purchase from seller s_j is c. Also there is a retail price vector $P_j = (p_{j0}, p_{j1}, \ldots p_{j,m-1})$ for each seller s_j. If seller s_j has no good g_k available, $P_{jk} = 0$. The objective of the problem is to minimize the cost to each buyer in B. To solve this problem and evaluate the performance, we need to define the following terms.

2.2. Definition of Terms

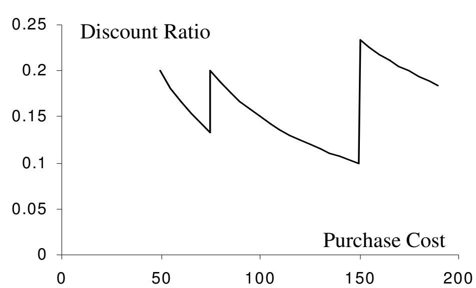

Figure 1: The Discount Ratio Function

Discount Ratio: The discount ratio is defined as the ratio of the discount to the corresponding cost. The discount ratio must have an upper bound in a real market, because sellers need to guarantee their profit ratio. Hence, searching the maximum discount could be interpreted as finding the highest discount ratio the buyers can.

The discount ratio function may not be monotonic non-decreasing, meaning that more cost is not necessary to cause higher discount ratio. For example, J.C. Penney provides $10 off for purchases over $50, $15 off for purchases over $75 and $35 off for purchases over $150 on a certain day. Figure 1 shows the corresponding discount ratio function.

Utility of a Buyer: The utility of a buyer is defined as the difference between the discount he can obtain by shopping alone and the one he can get by using a purchase strategy based on both bundle search and buyer coalition formation. Buyers try to maximize their own individual utility.

Coalition: A coalition (*CL*) is a subset of the buyer set *B*. Members of a coalition will contribute at least a portion of their total cost of buying all goods that they need.

Coalition Structure: A coalition structure (*CS*) is defined as a partition of the agents in a system into disjoint coalitions in prior work [12,13]. Because each buyer needs to buy multiple goods, we allow a buyer to join multiple coalitions simultaneously by contributing some of his purchase items. A coalition structure is not a partition of buyers, but a partition of all goods that all buyers need to buy.

Coalition Value: The coalition value of a coalition *CL* is defined as the sum of the utilities that all members obtain through joining the coalition.

Value of Coalition Structure: The value of a coalition structure *CS* is defined as the sum of the values of all coalition in the coalition structure. The value of the coalition structure is equal to the sum of utilities of all buyers.

In this paper, we do not have to develop a centralized payoff division mechanism for the satiability of coalitions. Since the discount ratio of each transaction is fixed, a buyer's payoff by joining a coalition is the difference between the cost to the buyer by shopping individually and the cost by joining the coalition. It is possible for some buyers in the optimal coalition structure to spend more than those in another non-optimal coalition structure. We allow buyers to refuse to join a coalition that causes higher costs than joining another coalition would cause. So the objective of this problem should be translated into maximizing the value of coalition structure as well as the resulting coalitions should be stable in the core [8,11,13], which means that any subset of buyers in a coalition can get at least as much by joining the coalition as the value of the coalition formed by the buyers in this subset.

2.3. Assumptions

- Buyers are self-interested. Their purchase decisions depend on whether they can maximize their own individual utility.
- Buyers know all retail prices of items in *G* and the discount functions of sellers in *S*.
- Buyers do not know other buyers' purchase strategies and goals in advance.
- Buyers do not bargain with sellers. The discount function of each seller is fixed.
- Each buyer can join multiple coalitions at the same time if necessary.
- No discount is caused by partnership among sellers. If a partnership exists between a pair of sellers, we can consider them as one seller without losing generality.
- Since the discount is a function of the total cost of all goods sold in one transaction, we can view multiple same goods items as multiple different items. Without losing generality, we assume in each Q_i, q_{ij} is equal to 1 or 0.

3. Traditional Centralized Approach

Traditionally, the above purchase problem can be solved by a centralized approach. Suppose there is a buyer leader who has all information about buyers and sellers. The buyer leader searches for the optimal strategies for buyers. The simplest centralized approach is to enumerate all possible coalition structures of all goods that buyers need to purchase and find the optimal coalition structure that minimizes the costs of all buyers.

However, the computational cost of this approach is prohibitively expensive. The total number of all goods that all buyers in B need to purchase is given as follows:

$$NQ = \sum_{i=0}^{m-1} \sum_{j=0}^{l-1} q_{ij}$$

The total number of all possible coalitions is $2^{NQ} - 1$. Sandholm et al [12] have already proved that the total number of all possible coalition structures is $O(NQ^{NQ})$, which is so huge that not all coalition structures can be enumerated unless the number of all goods is extremely small (below 15 or so in practice). Also, for each coalition, we need to do an optimal bundle search, of which the time complexity is $O(N^M)$ in the worst case, where M is the number of goods, and N is the number of sellers [6].

Another costly computation of this centralized approach is evaluating whether the resulting coalitions are stable in the core [8, 11]. The reason to do this is because buyers will refuse to join a coalition that causes higher cost than the cost of joining other coalitions. Buyer leader needs to calculate the total cost to each buyer in every coalition structure and find whether the coalitions are stable [8,11].

In real Electronic Markets, buyers are self-interested and geographically distributed. They make purchase decisions based on their local information and on minimizing their own cost. The incentive of buyers to join a buyer coalition is to obtain a greater discount than they would from purchasing individually. It is more realistic to let buyers make their own decisions [9, 10] and form coalitions through negotiation than set a coalition formation leader to evaluate the coalition value and distribute the payoff. We propose a distributed approach to solving the purchasing problem. It is much more efficient and practical than the centralized approach for real applications.

4. Distributed Approach

The basic idea behind our approach is that a buyer makes his own decision based on maximizing his own utility. The coalition formation depends on negotiation among buyers instead of any mediation by a group leader. Hence, solving the purchasing problem turns into designing a mechanism that causes each buyer's decision to achieve both local and global optimality. Furthermore, as we mention before, the discount ratio from one seller in each transaction has an upper bound. Whenever a buyer finds that the discount he can obtain from one seller has reached the upper bound of the discount ratio, he can make a decision immediately without considering other possibilities.

Based on the above analyses, our approach to solving the purchasing problem needs two steps. At first, buyers do their individual bundle searches to find the optimal bundle for their own shopping lists. If the discount ratios obtained from the sellers involved in the optimal bundle are the maximal discount ratios that the sellers can offer, buyers do not have to form or join any buyer coalition to increase the amount of discount they can gain. Otherwise, buyers start the second step of searching for coalitions proposed by other buyers or proposing new coalitions to related buyers.

The reason for a buyer to do bundle search first is that the utility resulting from the individual bundle search belongs to the buyer for sure. The possible utility obtained from joining a buyer coalition is uncertain because it depends on whether buyers can achieve a consensus. The buyer needs to guarantee certain amount of discount. It is possible to lose the optimal result by separating the bundle search and the coalition formation into two steps. But it ensures that buyers obtain the discount at least as much as they can get individually.

To propose a new coalition, the buyer needs to know who else is interested in joining a buyer coalition. In this paper, we assume no discount caused by partnership among sellers. Hence, the objective for buyers to join a coalition is to obtain a greater discount from one seller. We set an independent buyer club agent for each seller in S. If buyers are interested in joining the coalitions related to a specific seller, they register in its buyer club and obtain information about other buyers who need to join coalitions from the buyer club. Buyers and buyer club agents communicate through exchanged messages.

Buyers start their coalition search based on the results of their bundle search in the first step. Each buyer only registers with the buyer clubs involved in the individual optimal bundle result. When buyers' bundle searches do not include a particular seller, or when they have already obtained the optimal discount ratio from a seller, they do not register with the corresponding buyer clubs.

4.1. Bundle Search Problem

The bundle search problem for each buyer b_i is defined as following: Given a shopping vector $Q_i = (q_{i0}, q_{i1}, \ldots, q_{i,l-1})$, where q_{ij} is equal to 1 or 0. There is a set of sellers $S = \{s_0, s_1, \ldots, s_{n-1}\}$ who can supply partial or all goods in Q_i. Each seller s_j ($j = 0, 1, .., n-1$) has its own discount function $\delta_j(c)$: $R^+ \to R^+$. The problem is to find an optimal purchasing strategy, i.e., buy goods in Q_i from a subset of sellers in S, who can provide minimal cost to purchase all goods with $q_{ij} = 1$ in Q_i for buyer b_i.

The optimal algorithm for the bundle search problem is the Full Cartesian Algorithm [6]. The idea is to enumerate all combinations of sellers for all goods on the buyer's shopping list. The combination that results minimal cost is the optimal bundle. The time complexity of this algorithm in the worst case is $O(N^M)$ [6]. If the prices of items on the shopping list are not affected by adding more items, we can use dynamic programming to solve this problem with time complexity $O(N^2)$. However, this is not the case for the purchasing problem in this paper. The price of each item is changed by the discount ratio that the buyer obtained. The discount ratio is a function of the total cost to a buyer. If we use a dynamic programming algorithm, when a new item is added, the total cost may change and the discount ratio may also change. Then the price of items that have already been calculated may change. Hence, the dynamic programming algorithm is not appropriate for this problem.

We have developed a very efficient heuristic algorithm to solve this bundle search problem. It is called Maximal Gain Bundle Search (MGBS) algorithm that described in Figure 2. The algorithm is based on the following three heuristic rules:

Rule #1 Maximal Bundle: The problem of bundle search comes from the general economic case, the more one spends with a single seller, the more discount one gets from that seller.

Maximal bundle means buying as many items from one seller as possible. If the cost of a bundle with the maximal discount from every available seller is larger than the sum of the corresponding minimal retail prices for the goods in the bundle, then it is not necessary to continue the bundle search. The buyer just needs to buy all goods in the bundle at the lowest retail prices.

Rule #2 Maximal Gain Ratio: If the costs of maximal bundles from multiple sellers are less than the sums of the corresponding minimal prices, we pick up the seller with the maximal "*Maximal Gain Ratio*" as the candidate seller.

Maximal gain ratio is defined as a ratio of the difference between the sum of minimal retail prices of the bundle and the cost of the bundle after applying a discount to the sum of minimal retail prices of the bundle. Suppose there is a set of sellers, $Sb = (Sb_0, Sb_1, \ldots, Sb_k)$ for a bundle of goods, $Gb = (Gb_0, Gb_1, \ldots, Gb_k)$. The gain ratio $g(Gb, Sb)$ of the bundle of Sb is defined by the following equation:

$$g(Gb, Sb) = \frac{\sum P_{\min}^{Gb} - (\sum P_{Sb}^{Gb} - \sum D_{Sb}^{Gb})}{\sum P_{\min}^{Gb}}$$

$\sum P_{Sb}^{Gb}$ denotes the sum of the prices of all goods in Gb of the sellers in Sb. $\sum P_{\min}^{Gb}$ denotes the sum of the minimal prices to purchase all goods in Gb. $\sum D_{Sb}^{Gb}$ denotes the sum of the discounts obtained from all of the sellers in Sb for purchasing goods in Gb as a bundle.

Rule #3 Bundle Regression: Since the discount ratio could be non-monotonic increasing. Through the inverse function of the discount function, we can find the minimal cost to get the same amount of discount of a maximal bundle. Based on this minimal cost, we search for the cheapest bundle purchase with the same amount of discount from this seller, and leave the other goods for another round of searching. Before calculating the maximal gain ratio, the maximal bundle for each seller should be refined to the *minimal bundle* from the seller with the same amount of discount as the maximal bundle. This rule provides a method to refine the search results already obtained from the two rules above. The heuristic goal here is to achieve a higher discount ratio for each partial bundle purchase.

```
Maximal Gain Bundle Search Algorithm: {
  DoneSearch = false;
  G = all goods;
  While (!DoneSearch) {
    Find the maximal bundle mb for G;
    If (mb ≥ the sum of minimal retail prices of G)
    Then { DoneSearch = true; }
    Else {Find the seller S_x with Maximal Gain Ratio;
          Do bundle regression for S_x;
          Remove goods as much as possible from G such
          that the buyer still gains same discount;
          If (G = ∅)
          Then {DoneSearch = true;}}
  }
}
```

Figure 2: Maximal Gain Bundle Search Algorithm

The outputs of the bundle search for buyer b_i is a seller vector $SV_i = (sv_{i0}, sv_{i1}, \ldots, sv_{i,l-1})$, where sv_{ij} = null if $q_{ij} = 0$, otherwise sv_{ij} is equal to the corresponding seller's ID. With the seller vector, buyer b_i can calculate whether he has obtained the highest discount ratios from sellers in SV_i. For those sellers from whom the buyer b_i does not obtain the highest discount ratios, he sends buyer partner request messages to the corresponding buyer club agents. A buyer could join multiple buyer clubs, but the coalition formation of a buyer club is totally independent from those of other buyer clubs.

4.2. Distributed Coalition Formation through Explicit Negotiation (DCF-EN)

In our distributed coalition formation mechanism, buyers form coalitions through explicit negotiation (denoted by DCF-EN mechanism). There are three main issues in defining a negotiation mechanism [2]: the space of possible deals, the negotiation process, and the negotiation strategy.

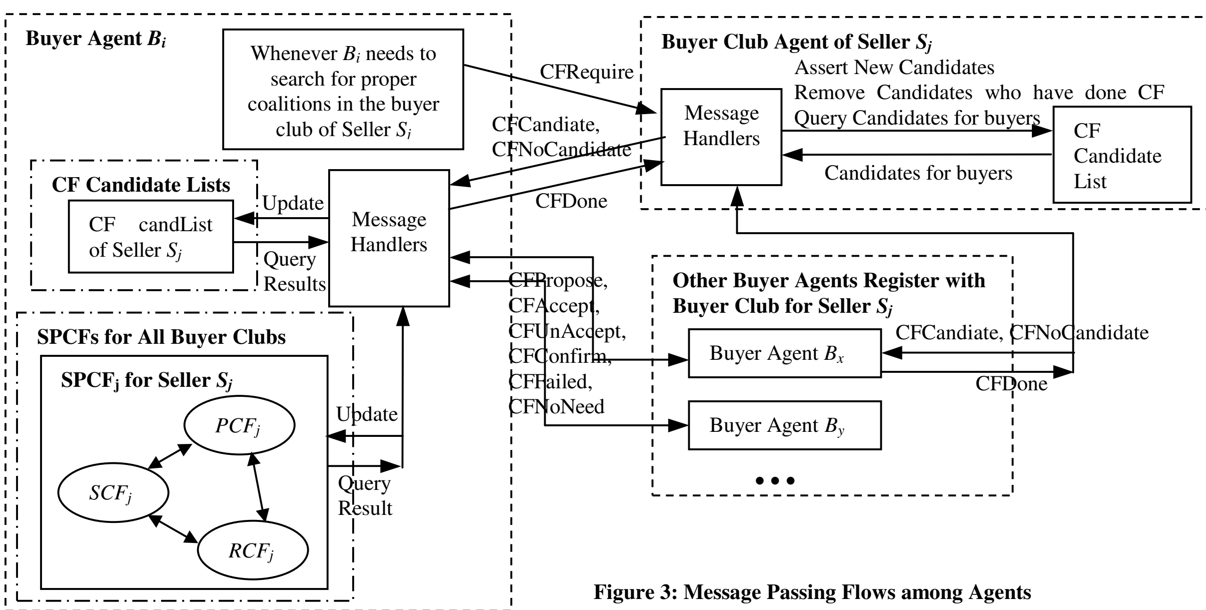

Figure 3: Message Passing Flows among Agents

In our negotiation protocol, all possible coalitions construct the space of possible deals (*SPCFs*) for a buyer. A buyer can have multiple *SPCFs*, one SPCF for one buyer club. Any one of his *SPCFs* includes three kinds of possible coalitions for the corresponding buyer club: the proposals he has sent out (*SCF*); the proposals he received (*RCF*); and all other possible proposals (*PCF*) that are not in SCF and RCF. The relationship among the above coalition sets is given by the following two equations:

- $SPCF_x = SCF_x \cup RCF_x \cup PCF_x$;
- $SCF_x \cap RCF_x \cap PCF_x = \varnothing$.

where, x refers to the index of the buyer club.

Buyers negotiate with each other through sending messages. Figure 3 shows all message flows among buyer agents and buyer club agents. The negotiation process for a buyer is to process negotiation messages from outside of a buyer. Any negotiation decision is made based on the negotiation strategy of buyers. We call a message handling method "message handler". Figure 4 shows all message handlers of a buyer agent in the DCF-EN mechanism.

In our current DCF-EN mechanism, all buyers use the same negotiation strategies:

- Each buyer can propose multiple coalitions without waiting for the confirmation of the coalitions that have been sent out. However, all these coalitions that has been sent out parallel should cause same utility for the buyer at any given time.
- All coalitions that have been sent out in parallel should cause same utility for the buyer at any given time.
- Each buyer who receives multiple coalition proposals can only accept to join one coalition at any given time (no regret decision [15]).
- Buyers greedily accept the best coalitions they can get. However, if their utilities derived from joining a coalition or from opting out [15] are the same, they will not join it.
- Each buyer terminates his negotiation process due to buyer coalitions formed (accepted), failed (refused), or time out.

CFCandidate from the buyer club agent of S_j

If (CFSearch for S_j not done)
Then {
  Assert new *CF* candidate to *CF* candidate list for seller S_j;
  Update *CF* space for the buyer club of seller S_j;
  If (there exists a new optimal coalition)
  Then {send CFPropose to the corresponding buyer agents and store it in SCF_j;}
  Else {store new *CFs* in PCF_j}
}

CFAccept from a buyer agent of S_j:

If (CFSearch for S_j not done)
Then {
  If (All members in the *CF* have accepted And not send any CFAccept)
  Then {Send CFPropConfirm to all other buyer agents in the *CF*;
    Send CFDone to the buyer agent for S_j;
    Send CFNoNeed to all buyer agents in the Candidate List for Sj who are not members in the *CF*;
    If (The size of SCF_j > 1))
    Then {Send CFPropFailed to all members of the other *CFs* in SCF_j}
  }
}
Else {Send CFNoNeed to the sender}

CFAccept from a buyer agent of S_j

Send CFDone to the buyer agent for S_j;
Clean its *CF* space for S_j;
End its *CF* search process for S_j.

CFNoCandidate from the buyer club agent of S_j

If (CFSearch for S_j not done)
Then {
If (SCF_j is empty)
Then {
  If (has not sent CFPropAccept)
  Then {
    Select the best current *CF* in *CF* space for seller S_j;
    If (the best CF is in RCF_j)
    Then {send CFPropAccept to the proposed buyer agent}
    Else {Send CFPropose to the members in the best *CF*;
      Update PCF_j and SCF_j }
  }
}}

CFFailed from a buyer agent of S_j:

If (CFSearch for S_j not done)
Then {
  If (Has accepted the *CF*)
  Then{Allow Bi to accept *CF* again;}
  If (No more new candidates for S_j)
  Then {Remove the *CF* from RCF_j;}
}

CFUnAccept from a buyer agent of S_j:

If (CFSearch for S_j not done)
Then { Allow Bi to accept CF again;
  Send CFPropFailed to other members in the *CF*;
  If (No more new Canidates for S_j)
  Then {Remove the *CF* from SCF_j.}
}

CFPropose from a buyer agent for S_j

If (CFSearch for Sj not done)
Then {
 If (the *CF* not in RCF_j, SCF_j and PCF_j)
 Then { If (the *CF* is optimal)
    Then {Send CFPropAccept}
    Assert the *CF* to RCF_j.
 }
 If (the *CF* in SCF_j)
 Then {If (has not sent CFPropAccept
    Then {Send CFPropAccept}
 }
 If (the CF in PCF_j)
 Then {Remove it from PCF_j;
    Assert it to RCF_j;
    If (the *CF* is optimal)
    Then {Send CFPropAccept}
 }
}
Else {send CFNoNeed to the sender}

CFNoNeed from a buyer agent of S_j:

If (CFSearch for S_j not done)
Then {
 For all buyer agents in the message {
  Remove *CFs* in PCF_j, which include the buyer agent;
  If (Has accepted a *CF* in RCF_j, which include the buyer agent)
  Then {Allow B_i to accept *CF* again}
  Remove *CFs* in RCF_j, which include the buyer agent;
  Send CFPropFailed to all members in *CFs* in SCF_j, which include the buyer agent;
  Remove *CFs* in SCF_j, which include the buyer agent;
 }
}

Figure 4: Message Handlers for a Buyer Agent B_i

```
Buyer Club Agent Algorithm: {
    Start the message listener;
    While (True) {
    If there are new messages that have not processed,
    Then {Get the message which arrived earliest;
            Call the corresponding message handle
            method based on the type of the message;
            }
    For each buyer b_x in the candsList {
        If size(candsSent_x) < size(candsList) –1
        Then {Send rest candidates to b_x;
                Add the candidates sent to candsSent_x; }
    }  where candsList is the coalition candidate list of
        this buyer club and candsSent_x is the candidates
        sent vector of the buyer b_x.
    If there are no more new candidates,
    Then {Send CFNoCandidate to all in candList; }
    }
}
```

Figure 5: Buyer Club Agent Algorithm

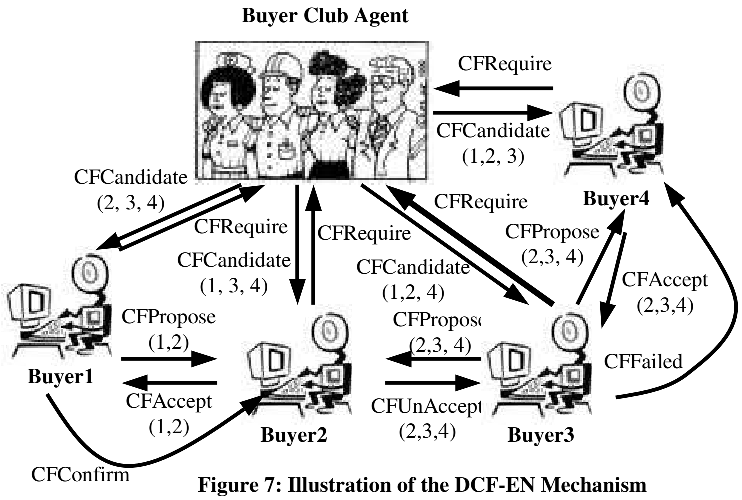

Figure 7: Illustration of the DCF-EN Mechanism

```
Buyer Agent Algorithm: {
  DonePurchase = false;
  Start the message listener:
  Run the bundle search algorithm to get the optimal
  bundle with a seller vector SV = (sv_0, sv_1, …, sv_{l-1});
  For each seller in SV {
    If (the cost in the optimal bundle does not cause
      the seller offers its highest discount ratio)
    Then {Send CFRequire to the corresponding buyer club agent;}}
  If (all sellers in SV do not need to send CFRequire)
  Then {DonePurchase = true;}
  Else {
    While (True) {
      If (there are new messages that have not been processed)
       Then {Get the message which arrived earliest;
        Call the corresponding message handler based on
          the type of the message; }
      If (DonePurchase != true)
      Then {
        For all buyer clubs {
         If (the buyer does not finish coalition search;
            and no more new candidates;
            and has not sent CFAccept;
            and the corresponding SCF is empty.)
          Then {Find the current best coalition bc;
                If (bc is from RCF)
                Then {send the buyer who proposed CFAccept }
                Else {propose the coalition to other buyers;
                      Remove bc from PCF and add it into SCF}
         If (the buyer finishes all coalition formation process)
         Then {DonePurchase = true.}
}}}}
```

Figure 6: Buyer Agent Algorithm

4.3. Complete Algorithms

To implement our DCF-EN mechanism, we need to implement two kinds of agents. The buyer agent algorithm is given in Figure 6.

Buyer club agents do not involve in negotiation directly. They only provide coalition candidates information to buyer agents. Figure 5 gives the algorithm of buyer club agent.

Figure 7 illustrates how the DCF-EN mechanism works in an example. In this case, there are four buyers who send coalition formation requirement to the buyer club agent. The buyer club agent sends the corresponding candidates to each buyer.

Before each buyer sends out or accepts any proposals, he constructs his *PCF* for each seller by calculating the discount ratios he can obtain through joining all possible coalitions (To simplify the implementation, we currently let a buyer agent contribute his total cost for one seller to one buyer coalition. Theoretically, a buyer can split his cost for one seller to multiple coalitions to obtain a better discount, but the computational complexity would be awfully intractable). In this case, all possible coalitions for each buyer are totally $2^3 = 8$ (we do not consider an individual buyer as a coalition here). If a discount ratio is larger than the discount ratio that he can obtain by shopping alone from the corresponding seller, he stores the corresponding coalition into his *PCF* for that seller.

After a buyer finish constructing his *PCF* for a seller, if it is not empty, he picks up the best possible coalition and sends CFPropose to the corresponding buyers. Or if he has already received a proposal that is the same as the best possible coalition, he sends back to the buyer who proposed CFAccept and waits for the message of CFConfirmation or CFFailed.

Suppose that the best possible coalition for Buyer1, Buyer2, Buyer3 and Buyer4 are (Buyer1, Buyer2), (Buyer2, Buyer1), (Buyer3, Buyer2, Buyer4) and (Buyer4, Buyer2, Buyer3) respectively. Buyer1 and Buyer3 send CFProposes to Buyer2 and Buyer4 first. After Buyer2 receives the CFPropose(1,2) from Buyer1, it accepts that and sends CFUnAccept to Buyer3 because Buyer3 sends CFPropose(2,3,4). After Buyer3 accepts CFUnAccept, it sends CFFailed to Buyer4 because Buyer4 sent CFAccept before. Buyer1 sends CFConfirm to Buyer2 to confirm their coalition. Then Buyer1 and Buyer2 end their searching process. Buyer3 and Buyer4 may continue their searching processes if they can find other partners to form coalitions such that they can obtain better discounts than they purchase alone.

Each buyer sends the deadline requirement for the coalition candidates to the buyer club agent when he registers for searching for buyer coalitions. The buyer club agent will send back to the buyer all candidates who register before the deadline. After that, it sends CFNoCandidate. Whenever a buyer sends CFDone, the buyer club agent will remove it from his candidate list. Due to the communication delay, this cannot guarantee that the candidates sent to a buyer agent are not obsolete. Hence, we let the obsolete buyer agents send CFNoNeed to the buyer who proposes to them after they have finished their coalition formation process.

Another important temporal parameter in DCF-EN mechanism is the expecting end time for a coalition proposal, which is important for the following two reasons: Buyers should have deadlines for their purchase goals; Since every buyer can propose possible coalitions, there could exist deadlocks when buyers are waiting for confirmations for different proposals that involve a same buyer. The expecting end time can avoid such a deadlock.

Indeed, the real implementation of the DCF-EN mechanism is much more complex than what Figure 7 shows. For example, if Buyer1 and Buyer2 send their best possible coalition proposals at to each other simultaneously, what will happen? If one buyer receives multiple proposals at same time, what should he do? How does each buyer update his *SPCFs* whenever he receives new proposals or its proposal gets refused? What if the proposal he receives is the same as the one he has sent out? How does a buyer agent handle the asynchronous issues? How does the order of messages a buyer agent received affect the negotiation process? And so on.

In one word, how does the automated multi-party negotiation run properly without a centralized mediator? The pseudo code of the message handlers in Figure 4 gives the technical details how the DCF-Mechanism handles above issues.

4.4 Complexity and Correctness Analyses

A buyer finds his own optimal bundle strategy first. If we use the exhaustive bundle search algorithm, the worst case time complexity is $O(N^{M'})$, where *M'* is the number of items that the individual buyer needs to buy. N is the number of sellers. If we use the MGBS algorithm, the time complexity in the worst case is *O(CNM')*. *C* is the times of iterations and *C<M'*. The upper bound of the total cost to a buyer with the MGBS algorithm is the sum of the minimal retail prices of all goods that the buyer needs to buy. Each buyer only needs to do a bundle search once.

Since the coalition formation process for each buyer club is independent of other buyer clubs, buyers do not have to do bundle searches during the coalition formation process. The *SPCFs* of a buyer only includes coalitions that include him.

In the worst case, suppose all buyers need to buy all goods from one seller, for each buyer b_i, the number of goods he needs to buyer is $NQ_i = \sum_{j=0}^{l-1} q_{ij}$. The total number of coalitions that the buyer b_i needs to evaluate is $O(2^{NQ-NQ_i})$.

Compared to the traditional, centralized approach, our distributed approach significantly reduces the complexity of the searching process, even while using exhaustive search algorithms.

The classical core is the strongest of the solution concepts in coalition formation [13]. The core of a game is a set of payoff configurations, where each $\vec{x}$ is a vector of the payoff *P(CL)* of a coalition *CL* in a coalition structure *CS* to the agents, in such a manner that no subgroup is motivated to depart from the *CS*. The purpose of this concept is to maximize social welfare (group rationality) and to motivate agents to stay with the social welfare maximizing coalition structure (individual rationality). Furthermore, every subgroup of agents in coalition is better off staying within this coalition than forming a coalition of their own (coalition rationality) [13].

The concept is so strong that the core of a coalition game can be empty in many cases [13]. In this paper, we relax the classical core concept to emphasize only the coalition rationality. That is, after a coalition formed, no subgroup of the coalition is willing to form its own coalition. We define a coalition is stable in the core in terms of coalition rationality as the following:

Definition: Given $\vec{x}$, a vector of the payoff *P(CL)* of a coalition *CL* which is composed of a set *A* of agents. We say *CL* is stable in the core in terms of coalition rationality if and only if agents in any subset *A'* of A can get at least as much by joining the coalition *CL* as they obtain by joining the coalition formed by the agents in this subset *A'*.

No buyer needs to consider evaluating the value of the resulting coalition structure in our DCF-EN mechanism. We have the following claim:

Claim: The coalitions formed through the DCF-EN mechanism are stable in the core in terms of coalition rationality.

Proof: Each buyer always tries to join the best coalitions that he can find in our approach. The best coalition to a buyer is the one that maximizes his own utility. Hence, any coalition that has been accepted by all its members must be the best coalition for all members that they can find. The values of subset coalitions of this coalition cannot be better than its value. □

As for the negotiation mechanism, the communication load is not high at all for the following reasons: Buyers will not send out a coalition proposal unless they find the coalitions that cause the highest discount ratio or they finish building their search space and finds the best coalition he could join. Meanwhile, before buyers finish building his whole search space, they only accept the coalition proposals that cause the highest discount ratio. A buyer agent only needs to send available candidates to buyers who need to join coalitions once.

5. Simulation Results

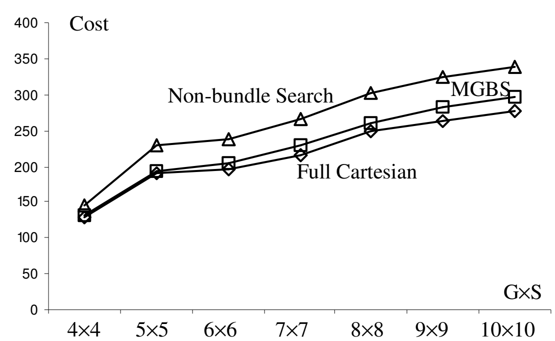

Figure 8: Results of Bundle Search Algorithms

First, we show the simulation result of the MGBS algorithm for the bundle search problem. Figure 8 shows the total costs of buying n kinds of goods that can be provided by n different sellers (G×S matrix) through different algorithms. Buyers have different shopping lists and the retail prices offered by different sellers are different. By a non-bundle search algorithm, buyers only buy all goods with the minimal retail prices from n sellers. Full Cartesian generates optimal results for the bundle search problem. The simulation results of the MGBS algorithm are close to the optimal results. The MGBS algorithm is much more efficient than the Full Cartesian algorithm with more sellers and goods as the simulation result showed in Figure 9. It is hard to show the execution times of 7×7, 8×8, 9×9, and 10×10, because the magnitude is too high compared with the execution time of MGBS algorithm.

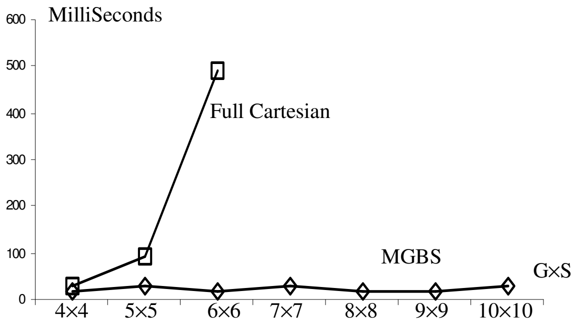

Figure 9: The Execution Time of Different Algorithms

Table 1: Parameters Used for Simulation

Parameters	Experimental Values
N×N (G×S matrices)	3×3, 4×4, 5×5
Numbers of Buyers	3, 4, 5, 6, 7, 8, 9
Discount Function	$F(c) = \begin{cases} 10, \text{ if } 50 \le c < 100; \\ 20, \text{ if } 100 \le c < 150; \\ 35, \text{ if } 150 \le c < 200; \\ c \times 20\%, \text{ if } c \ge 200. \end{cases}$

Based on the purchasing problem definition in Section 2, we give the input parameters and the discount function used in our simulation in Table 1. Since, in a real market, the buyers with a small amount of purchase cost are more likely join a buyer coalition, we did not use large G×S matrices in our simulation. Without losing generality, sellers have the same discount policies, but the retail prices offered by different sellers are different.

To evaluate the results of our solution to the purchase problem and the efficiency of the distributed coalition formation mechanism, we need to test the costs of buyers through different purchase strategy and the communication cost for each buyer. In our simulation, we evaluate the following parameters: the average cost of each buyers and the total cost of all buyers with different purchase strategies; the average total number of messages a buyer needs to handle and the average number of messages from one buyer to another buyer.

Since the computational cost is too high to run an optimal algorithm for the purchasing problem, to compare our results with the optimal results for a certain purchase problem, we use the lower bound of the optimal cost for a buyer, which is the sum of the minimal retail prices of all his goods with obtaining the highest discount ratio in the market. In real markets, it is impossible for buyers to obtain this lower bound cost.

Figure 10 shows the value of the resulting coalition structures with 3, 4, 5, 6, 7, 8, and 9 buyers by different purchase strategies. Figure 11 shows the average cost to each buyer. RetailMinPriceCost denotes the cost to a buyer using the strategy of only searching for the minimal retail price in the market for each item needed. OptimalBundleCFCost denotes the cost to a buyer of doing an optimal bundle search first and then trying to join buyer coalitions. MGBSBundleCFCost denotes the cost to a buyer of doing a MGBS bundle search first and then trying to join buyer coalitions. LBOptimalCost is the lower bound of the optimal cost that a buyer has to pay.

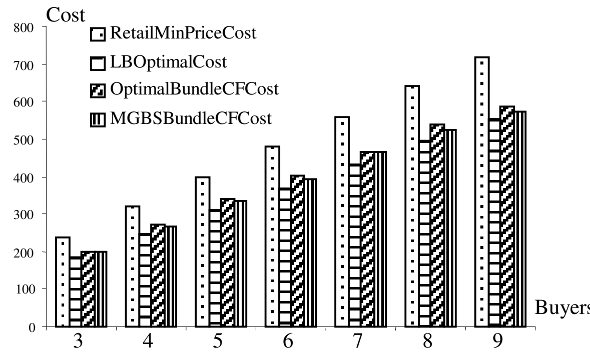

Figure 10: The Values of the Resulting Coalition Structures

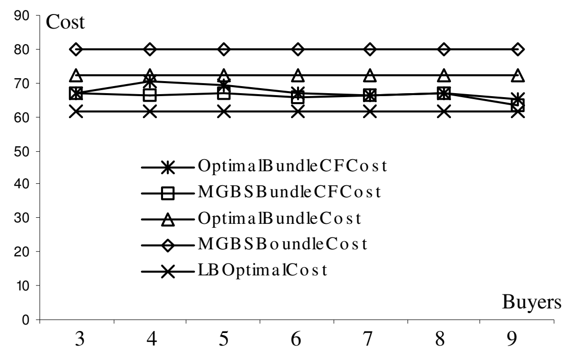

Figure 11: The Average Cost of Each Buyer

The MGBSBundleCFCost is close to the lower bound of the optimal value. Surprisingly, it is lower than the OptimalBundleCFCost in general. The average cost of each buyer is not increasing or decreasing significantly along with the number of buyers increasing in the market. This is because buyers are not interested in joining large-scale coalitions but the most profitable coalitions.

OptimalBundleCost refers to the cost to a buyer of doing an optimal bundle search without joining any buyer club. MGBSBundleCost refers to the cost to a buyer of doing a MGBS bundle search without joining any buyer club. Figure 11 shows that combining bundle search strategy and buyer coalition formation strategy can reduce the cost more than just doing a bundle search and the cost is very close to the optimal cost.

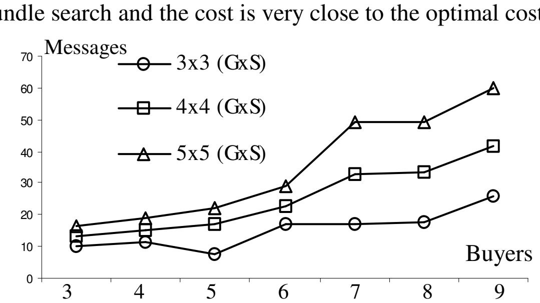

Figure 12: All Messages Received by One Buyer

Figure 12 and Figure 13 show that the communication load during the entire coalition formation process is quite low. Interestingly, the number of messages from one buyer to another buyer are not increasing but decreasing sometime with the number of buyers increasing in the market. Also, this number increases slowly with the size of G×S matrix increasing. The reason is that buyers are not interested in joining large-scale coalitions but the most profitable coalitions as early as they can.

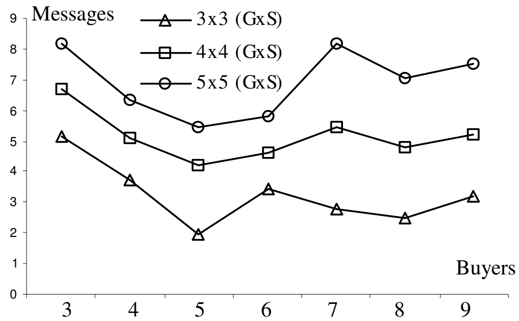

Figure 13: The Messages From One Buyer

The number of the messages sent from a buyer club agent to a buyer is equal to the number of buyers in the market. Because a buyer club agent needs to send all buyers in the buyer club a CFNoCandiate message. A buyer agent needs to send two messages to a buyer club agent. One is CFRequire and another is CFDone message.

Another important simulation result shows that our negotiation protocol can terminate the distributed coalition formation process for each buyer appropriately without a global controller.

6. Conclusion and Future Work

In Electronic Markets, buyers can easily access information about minimal retail price and discount policy of sellers with very low cost. Hence, it is valuable to build efficient purchase strategies for buyers who need to buy a small amount of goods and have no bargaining power [5]. Bundle search and buyer coalition formation are two profitable strategies for such small buyers. It is very valuable to combine these two purchase strategies together for a small buyer to obtain more discount based on the different discount policies of multiple sellers.

In real markets, there are many sellers who provide discount policies based on the total cost in each transaction. This type of discount policy allows buyers with different purchasing goals (even if they are not complementary) to form buyer coalitions to obtain greater discounts. The discount ratio offered by a seller must have an upper bound in a real market. Trying to find a purchase strategy to reach the highest discount ratio is a reasonable incentive for buyers to both do bundle searches and join buyer coalitions.

However, for the algorithmic aspect, both bundle search strategy and buyer coalition formation strategy are computationally intractable. Combining two of them together is extremely hard to compute the optimal purchase decision.

Combinatorial Coalition Formation [8] in electronic markets is a notoriously hard problem. Most related work on the CCF problem [1,7,8] is based an assumption, under which the price of commodity goods depends on the total number of the items sold in each transaction. The traditional mechanisms for the CCF problem in electronic markets are centralized [1, 8, 9]. Besides the high complexity, another difficulty arises in developing an appropriate payoff division mechanism that results in stable coalitions. Lerman and Shehory [9] developed a distributed buyer coalition formation mechanism for a large-scaled electronic market, but their model excludes an explicit negotiation protocol, and buyers encounter other buyers and coalitions randomly. They also assume that buyers need to purchase the same specific product.

In this paper, we present a distributed mechanism for buyers to use both bundle search and buyer coalition formation strategy in an electronic combinatorial market. To reduce the complexity, we developed an efficient heuristic algorithm MGBS for bundle search. The complexity of buyer coalition formation is reduced through distributing the coalition evaluation to each buyer who needs to join buyer coalitions. Without giving any centralized payoff division mechanism, the resulting coalitions are stable in the core in terms of coalition rationality [13]. We designed a negotiation protocol for coalition formation process with very low communication cost. Our mechanism can handle multiple coalition formation processes for buyers in parallel. Buyers can have different purchase goals.

The simulation results show that our approach to solving the purchasing problem defined in this paper is very practical and efficient. The resulting cost to buyers is close to the optimal cost. The communication load caused by the negotiation among buyers is very low. The negotiation processes terminate properly.

In our future work, it is possible to extend our DCF-EN mechanism to make it appropriate in a dynamic environment where buyers could join and leave randomly and the negotiation processes associate with time constraints.

An important issue that we need to consider in the future is the stability [2,14] of our negotiation mechanism. It is necessary to test whether other negotiation strategies that can cause better results for buyers. If buyers use different negotiation strategies, are the resulting coalitions still stable?

The DCF-EN mechanism can be extended to a general coalition formation mechanism. It could be applied to some other domains such as distributed resource and task allocation [10].

Our approach in this paper can be applied to some other domains. For example, service composition [16] is another popular research area in electronic commerce. It is the strategy of taking several component products or services, and bundling them together to meet the needs of a given customer. It is naturally related to the bundle search strategy. Our mechanism in this paper could be applied to developing strategies for service composition.

7. Acknowledgement

This work was supported in part by MURI grant #F49620-00-1-0326 from DoD and AFOSR.

8. References

[1] Ye, Y., and Tu, Y., Dynamics of coalition formation in combinatorial trading, in proceedings of International Joint Conference of Artificial Intelligence (IJCAI), 2003.

[2] Rosenschein, J. S. and Zlotkin, G., Rules of Encounter: Designing Conventions for Automated Negotiation among Computers, The MIT Press, 1994.

[3] Caillou, P., Aknine, S. and Pinson, S., A Multi-Agent Method for Forming and Dynamic Restructuring of Pareto Optimal Coalitions, in proceedings of the First International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS), 2002.

[4] Brynjolfsson, E., and Bakos, Y., Bundling information goods: pricing, profits and efficiency. Management Science, Dec. 1999.

[5] Proter, M. E., Competitive Strategy, The Free Press, New York, 1980.

[6] Chang, Y., Li, C., and Smith, J. R., Searching dynamically bundled goods with pairwise relations, in proceedings of ACM Electronic Commerce, 2003.

[7] Yamamoto, J. and Sycara, K., A Stable and Efficient Buyer Coalition Formation Scheme for E-Marketplaces, in proceedings of the Fifth International Conference on Autonomous Agents, 2001.

[8] Li, C. and Sycara, K., Algorithm for combinatorial coalition formation and payoff division in an electronic marketplace, in proceedings of the First International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS), 2002.

[9] Lerman, K. and Shehory, O., Coalition formation for largescale electronic markets, in proceedings of the International Conference on Multi-Aent Systems, 2000.

[10] Shehory, O. and Kraus, S., Feasible formation of coalitions among autonomous agents in non-super-additive environments, Computational Intelligence, vol. 15(3), 218-251, 1999.

[11] Kahan, J.P. and Rapoport, A., Theories of Coalition Formation, Lwarence Erlbaum Associates, Inc., London, 1984.

[12] Sandholm, T., Larson, K., Andersson, M., Shehory, O. and Tohme F., Coalition structure generation with worst case guarantees, Artificial Intelligence Journal, Vol (111), 209-238, 1999.

[13] Sandholm, T., Negotiation among Self-Interested Computationally Limited Agents, Ph.D. Dissertation, 1996.

[14] Sen, S. and Dutta, P. S., Searching for optimal coalition structures, in proceedings of the Forth International Conference on Multiagent Systems(ICMAS), 2000.

[15] Kraus, S., Strategic Negotiation in Multiagent Environments, The MIT Press, 2001.

[16] Preist, C., Bartolini, C., and Byde, A., Agent-based service composition through simultaneous negotiation in forward and reverse auction, in proceedings of ACM Electronic Commerce, 2003.