The Search for Coalition Formation in Costly Environments[1]

David Sarne[1] and Sarit Kraus[1,2]

[1] Department of Computer Science, Bar-Ilan University, Ramat-Gan, 52900 Israel
{sarned, sarit} @ macs.biu.ac.il
[2] Institute for Advanced Computer Studies
University of Maryland, College Park, MD 20742

Abstract. We study the dynamics of forming coalitions of self-interested autonomous buyer agents, for the purpose of obtaining a volume discount. In our model, agents, representing coalitions of various sizes, may choose to be acquainted with other agents, hopefully ending up with a joint coalition structure, which will improve the overall price. Upon encountering potential partnering opportunities for extended coalitions, the agent needs to decide whether to accept or reject them. Each coalition partnership encapsulates expected benefit for the agent; however the process of finding a potential partner is associated with a cost. We explore the characteristics of the agent's optimal strategies in the equilibrium and develop the equations from which these strategies can be derived. Efficient algorithms are suggested for a specific size-two variant of the problem, in order to demonstrate how each agent's computation process can be significantly improved. These algorithms will be used as an infrastructure from which the general case algorithms can be extracted.

1. Introduction

The growing interest in autonomous interacting agents has given rise to many issues concerning coalition formation. A coalition is a group of self-interested agents that agree to coordinate and cooperate in the performance of a specific task. Through the coalition, the agents as a group are able to perform their task more efficiently, and increase the participants' benefits [2, 6, 15, 16]. The main question in every coalition formation application is how to determine the set of agents each specific agent will be willing to form a coalition with. This is where each agent is associated with a specific type that captures special properties that characterizes it.

We consider environments in which an agent's utility is fully correlated with its type. The agent's type is additive and thus can be improved by forming coalitions with other agents. Agent types are ordered according to their associated utility. The higher the other coalition member's type the higher the agent's utility. Each agent in our model represents a coalition of one or more members. The agents may interact with each other to share information regarding their types. This information is used by each agent in its decision making process of whether to combine its current coalition with another agent's coalition, thereby forming a new coalition of a higher type. Consider, for example, the electronic marketplace where agents represent coalitions of buyers interested in a product. Assume the requested quantities determine the agent's type, thus the higher the requested quantity the better the price for the coalition members.

The agents' search for coalition opportunities is costly [10]: at each stage of its search an agent has to spend resources in locating and interacting with another agent representing a coalition of a random type. In addition, each stage of the search reflects a coalition coordination cost (communication with its members), which is derived from the number of coalition members.

The agent's willingness to extend the coalition upon encountering a new agent is insufficient. The new coalition will be formed only if it is mutually accepted by both

[1] This research was supported in part by NSF under grant #IIS-0208608.

agents' parties. A new agent will be created for the new coalition formed. This new agent will handle and represent all members previously handled by the agents that formed the new coalition. Each member in a given coalition will share the agent's costs relative to its type. It will also share the utility of the final coalition it will be a member of, relative to its type. Recalling that the types are additive and the agent's utility increases with its type, then a decision taken by an agent to form a coalition with another agent to improve its type, will never cause a conflict of interests with any of the members represented by this agent – the higher the increase in the agent's type, the more utility each member will be gaining.

Therefore, each agent, either created for a new coalition, or entering the marketplace as a representative of a single coalition member must consider two major questions: First, is it going to execute its task immediately or is it more beneficial to engage in costly search to extend the coalition? If the latter decision was taken then at each stage of the agent's search process, after reviewing the information regarding the current potential partner, the agent must make a decision whether to terminate the search or to continue. Continuing the search will hopefully yield a higher type partner to continue the process with (the agent will prefer rejecting partnerships reflecting significant future coalition coordination costs). Terminating the search will result in forming a coalition with the current potential partner if it agrees; this new coalition will face the same decisions, as mentioned above, all over again.

We utilize the electronic marketplace environment [16], as a framework for our analysis. The model we present considers buying agents, representing one or more different buyers, possibly interested in extending their coalition for buying a specific product. The benefit for all participants in such a coalition is in their ability to obtain, as a team, a discount price (compared to the price each of them would have paid separately). Each agent gains this utility separately, and the utility is not transferable (no side-payments). Obviously, the larger the quantity an agent is seeking to buy for the current coalition it is representing, the greater the benefit for its potential partners of an extended coalition and vice versa. Whenever a new coalition is created the new agent will seek to minimize the overall costs of such a coalition. The agent's utility (either by purchasing the product or by forming new coalitions) as well as its costs will be split among its represented members, according to their percentage out of the overall requested quantity. Therefore any decision the representative agent will take is the most beneficial for *all* its represented members.

The best price that can be obtained will be through one big coalition in which all agents are members. Yet, the introduction of search costs and coalition management costs into the model prevents this type of solutions, and enforces a genuine cost effectiveness analysis for evaluating each potential new coalition.

The same concepts of coalition formation through partnerships are valid and re-usable in other plausible MIS and CS related applications. Consider, for example, client-server environments, where distributed subroutines are waiting to be processed on a central server. Typically, the server processing time is a function of the query input. Here, there may be an incentive for a subroutine to partner with others to create a combined query for which the processing time over the server is shorter than the aggregated execution of each subroutine separately. Each subroutine's query characteristics can indicate a type, and the combined query possesses similar additive behavior as described above. Adoption of the proposed analysis and the suggested algorithms for such applications is simple once we express the search and coalition coordination costs associated with each application in terms of the coalition benefit.

Looking for a baseline for the coalition through partnerships problem we addressed AI and economic literature, as will be described in section 2. We continue in section 3 by presenting the general model for coalition formation through partnerships with search and coalition coordination costs. We show the general characteristics of equilibrium and develop the equations describing the agents' optimal strategies. In section 4 we utilize the basic two-size coalition variant to suggest algorithms for simplifying the distributed calculation performed by each agent. This is an important step towards extending the algorithms to handle the general case. We conclude and present directions for future work in section 5.

2. Related Work

Coalition formation processes focus a lot of attention on multi-agent systems [11, 13]. In recent years, research has introduced the coalition formation process also into electronic market environments. Recognizing the incentives for both buyers and vendors in volume discounts[2], several different buyer-agents coalition schemes were proposed [7, 12, 15, 16]. Extensions of the transaction-oriented coalitions to long-term ones, were also suggested [2, 3]. However, most research was mainly concerned with the procedures of negotiating the formation of coalitions and division of coalition payoffs. Other related popular research topics were finding the optimal division of agents into coalitions through a central mechanism and enforcement methods for the interaction protocols. The resources associated with agent's search for a coalition, and the influence of this factor over its decisions, were not discussed in this context[3]. Most mechanisms assumed an agent could scan as many agents as needed, or simply a central view of the environment.

The review of economic literature reveals that the model of partnerships with partner search costs was widely studied in traditional marriage markets and job-search applications, which evolved from the area of search theory [8, and references therein]. These models were focused on establishing optimal strategies for the searcher, assuming no mutual search activities and were classified under one-sided searches. In an effort to understand the effect of dual search activities in such models, the "Two-sided" search research followed. This notion was explored within the equilibrium search framework [4]. An interesting analysis through simulation of a two-sided market variant was introduced by Greenwald and Kephart [5].

Another important research area is the one called assortative matching. This area involves a decentralized search of more than two heterogeneous agent types. Becker [1] analyzed a costless matching market, where two different agent types produce a different utility when matched and otherwise no utility. Becker showed that the unique competitive equilibrium has assortative matching – meaning that matched partners are identical (in type), for both the transferable and non-transferable utility cases. Extended models which included some search cost elements for the non-transferable case were proposed by Smith [14] who modeled search "costs" by the discounting of the future flow of gains and Morgan [9] who used additive explicit search costs. The transformation of the suggested concepts in the economical models, into plausible applications over the internet and computerized environments with search and coalition coordination costs is not trivial. All the above economical models assumed no utility for an agent without forming a partnership, and most importantly, they didn't allow coalitions to extend themselves beyond two agents. We will refer to such models in section 4.

3. The model

We consider an electronic marketplace with numerous heterogeneous buying agents, representing different buyers. Each agent is characterized by the number of buyers it represents and the initial intention of buying a pre-defined quantity of a specific well

[2] The vendor agent was also claimed to possess several advantages for selling in bulk, mainly due to decreased advertisement costs and distribution costs.

[3] An exception can be found in [10], discussing settings where there are too many coalition structures to enumerate and evaluate (due to, for example, costly or bounded computation and/or limited time). Instead, agents have to select a subset of coalition structures on which to focus their search.

defined and easily found product. The encapsulated quantity of a coalition representative agent is an aggregation of all the quantities requested by the buyers this agent represents. The product price is a function of the purchased quantity - sellers offer a "discount for quantity" price, aiming to encourage purchase of large quantities in a bulk. Denoting the posted price for a quantity *q* as P_q, the price function satisfies the following:

$$\frac{dP_q}{dq} < 0 \quad , \quad \frac{d^2P_q}{dq^2} > 0 \qquad \textbf{(1)} \qquad\qquad \frac{d(qP_q)}{dq} > 0 \quad , \quad \frac{d^2(qP_q)}{dq^2} < 0 \qquad \textbf{(2)}$$

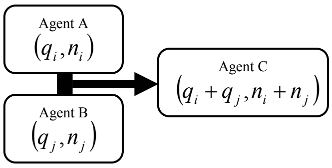
The first condition ensures the price per unit is a monotonically decreasing function (in a decreasing rate) of the requested quantity. This is mainly because sellers wish to create the incentive to buy wholesale. The second condition ensures basic economic principals of paying in overall more when buying more.

By introducing the above price function, and assuming that any quantity can be supplied, buying agents have an incentive to form a coalition with other buying agents in order to gain the price discount. However, finding a partner has its cost: for each stage of the search the process induces a specific search cost *a+bn*, where *n* is the number of buyers represented by the agent. The fixed cost *a* is related to the resources an agent spends on advertising its presence, locating other agents and interacting with them. The variable cost *b* is associated with the coalition coordination – the agent is required to maintain communication with the buyers it represents throughout the search process.

The agents are homogeneous in the sense that each agent has the same goal (to purchase a specific product at the lowest total cost), however they are heterogeneous in their types. We classify all the agents representing *n* buyers, who are interested in buying an overall quantity of *q* of the product, as agents of type *(q,n)*. *(*,k)* will be used to denote any agent representing *k* buyers and *(q,*)* will denote any agent requesting an aggregated quantity *q*. The *(q,n)* type agent can either engage in a search to extend its coalition, or buy the requested product, in its current configuration with an overall cost of qP_q. The agents are self-interested and therefore, given several alternatives, they will prefer to select the more beneficial ones.

Since the agent is not concerned with a limited decision horizon and the interaction with other agents doesn't imply any new information about the market structure, then its best search strategy for partners is sequential. Also, in spite of the existence of search costs, the agent's strategy is stationary – an agent of any specific type will not accept an opportunity it has rejected beforehand. At any stage of its search the agent randomly encounters one other agent interested in the same product. At the encounter, both agents will reveal their type (overall requested quantity and number of buyers represented by each agent). Then, each agent will make a decision whether to continue searching or to extend its coalition structure by partnering with the current encountered agent. In the latter case, the new coalition will take effect only if both agents are willing to form the partnership. Upon the creation of a new coalition, a new agent will be created, replacing the two agents in representing the joint coalition. The search and coalition coordination costs accumulated for each agent will be imposed on the buyers represented by it, relative to their requested quantity from the overall quantity. The new agent created, as any other agent in the marketplace, can either buy the product with a total cost of $(q_i + q_j)P_{q_i+q_j}$, or conduct a search to further extend its coalition with search and coordination costs of $a + b(n_i + n_j)$ - see Figure 1.

We start by assuming a continuous flow of new agents to the marketplace, each representing a single buyer. These agents are of type *(q,1)* where *q* is defined over the interval $[\underline{q}, \overline{q}]$. Since these are the basic bricks of future coalitions, the potential agent types can be described over the semi-continuous grid in Figure 2 (possible agent types are in gray).



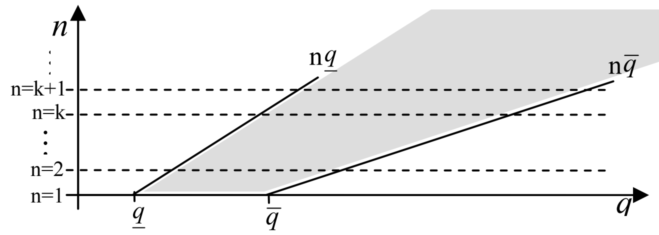


Figure 1 – Coalition Formation

Figure 2 – Possible coalitions

Notice that at this stage of the analysis, the gray area has no finite boundaries. This is simply because we consider all possible coalitions, regardless of equilibrium constraints and search profitability. We suggest that the area representing the agents' types engaged in search (of which we are interested in this paper) is actually bounded. Introducing search and coalition coordination costs into the model, we can easily set upper limits for this area. For each point q in the horizontal axis $q > \underline{q}$ we can find an agent type *(q,k)* for which it will be non-beneficial to engage in search, regardless of the distribution of other agent types currently engaging in search. This is where:

$$q(P_q - P_{q\to\infty}) = a + bk \quad \textbf{(3)}$$

Since the right side expression of (3) increases in *k*, then obviously all other agents of type *(q,n)*, where *n>k*, will prefer not to engage in search. Therefore the following expression (4) is an upper limit for possible agent types engaged in search in equilibrium:

$$n(q) = \min\left\{\left\lceil \frac{q}{\underline{q}} \right\rceil, \left\lceil \frac{q(P_q - P_{q\to\infty}) - a}{b} \right\rceil\right\},\ q \geq \underline{q} \quad \textbf{(4)}$$

A density function $f(q,n)$ is associated with each type of agents engaged in search and requesting a quantity q. We assume that agents, while ignorant of other individual agents' coalition sizes and requested quantities, are acquainted with the overall distribution of agent types in the market[4]. We also assume that this distribution is time-invariant[5]. Considering the population of agents engaged in search, we prove the following Theorem (1), for the agent's strategy in equilibrium.

Theorem 1. *(a) In equilibrium, an agent of type* (q_i, n_i)*, engaging in a search for potential partners, in order to extend its coalition, will use a reservation value strategy*[6] *according to a vector* $\vec{Q}_{q_i,n_i}$ *of quantity values (where* $Q_{q_i,n_i}[k]$ *represents the reservation quantity to be used when encountering a potential partner of type (*,k)). (b)* $Q_{q_i,n_i}[k+1] \geq Q_{q_i,n_i}[k]$.

Sketch of Proof:

(a) Assume that all agents are using a reservation quantity strategy and consider a (q,n) type agent which is willing to accept agent type (q_i, n_i) during the search. Obviously an alternate coalition with an agent (q_j, n_i), where $q_j > q_i$, will yield a better benefit. This can simply be achieved if the new agent created for the latter coalition will imitate the strategy of the one created for the first coalition (and be accepted by all the agents accepting the first one, according to the assumption). At a certain stage of the search, agent type $(q + q_i, n + n_i)$ will evolve to a coalition of type (q_k, n_k) which will prefer to buy the product than to extend the search. At this point, the original members of

[4] Using market indicators, spectator agents, etc..

[5] The method for maintaining a steady state population of agent type is important but will not be discussed in this context.

[6] A reservation value strategy is one where the searcher follows a reservation-value rule: It accepts all offers greater than or equal to the reservation value, and rejects all those less than this value.

coalition $(q+q_i, n+n_i)$ will be able to purchase the product at price P_{q_k} in comparison to price $P_{q_k+q_j-q_i}$ for original members of the coalition $(q+q_j, n+n_j)$.

(b) $Q_{q_i,n_i}[k+1] \geq Q_{q_i,n_i}[k]$ simply because the search costs of the new coalition, created with the agent representing more users, are higher. []

Notice that $Q_{q_i,n_i}[k]$ is a vector of a finite size as the area of potential agents to form a coalition with is finite (see equation (4) above). Denoting the expected cost of agent type (q_i, n_i), when using the optimal reservation vector $\vec{Q}_{q_i,n_i}$ as $V_{q_i,n_i}(\vec{Q}_{q_i,n_i})$, and the optimal cost of a coalition (q_i, n_i) as V_{q_i,n_i}, we obtain:

$$V_{q_i,n_i} = \min\{ V_{q_i,n_i}(\vec{Q}_{q_i,n_i}) \; , \; q_i P_{q_i} \}. \quad \textbf{(5)}$$

Using equation (5) we will be able to determine if an agent will engage in a search, according to its type. Consider agent of type (q_i, n_i) engaged in a search, at any given stage of its search. After reviewing the current potential partner's type (q_j, n_i), it has to make a decision whether to reject this partner and continue the search or partner with this agent in order to extend the coalition. Continuing the search in the current agent configuration will result in an expected future total cost of $V_{q_i,n_i}(\vec{Q}_{q_i,n_i})$. Accepting the partnership, will result in a future cost $(q_i V_{q_i+q_j, n_i+n_j})/(q_i+q_j)$ for agent (q_i, n_i) if the other agent agrees to form a coalition, or otherwise will enforce the agent to keep searching with an expected future total cost of $V_{q_i,n_i}(\vec{Q}_{q_i,n_i})$. Therefore, the optimal reservation value is the quantity $Q_{q_i,n_i}[n_j] = q_j$ where the agent is indifferent to the two options:

$$V_{q_i,n_i}(Q_{q_i,n_i}[n_j]) = V_{q_i,n_i}(\vec{Q}_{q_i,n_i}) \Pr(q_i < Q_{q_j,n_j}[n_i]) + \frac{q_i V_{q_i+q_j, n_i+n_j}}{q_i+q_j} \Pr(q_i \geq Q_{q_j,n_j}[n_i]) \quad \textbf{(6)}$$

Resulting in:

$$V_{q_i,n_i}(\vec{Q}_{q_i,n_i}) = V_{q_i,n_i}(Q_{q_i,n_i}[n_j]) = \frac{q_i V_{q_i+Q_{q_i,n_i}[n_j], n_i+n_j}}{q_i+Q_{q_i,n_i}[n_j]} \quad , \quad n_j = 1,2,... \quad (7)$$

The interpretation of equation (7) is that for any given agent of type (q_i, n_i), the overall cost when using its optimal reservation vector, $\vec{Q}_{q_i,n_i}$, equals its relative part (according to its quantity) in the overall cost of a coalition $(q_i + Q_{q_i,n_i}[n], n_i + n)$, for any *n*. This characteristic will play a key role in our further analysis.

In order to introduce the agent's search equations, we need to prove some additional characteristics of the equilibrium. First we will show consistency between agents' decision whether to engage in a search (Theorem 2). Then we will prove that for any two agents engaged in a search, the higher type agent (in terms of requested quantity) will use a higher reservation value for each *n* (Theorem 3). For this purpose first we need to prove Lemma 1.

Lemma 1. *The improvement in an agent's utility when buying the product in a coalition with any given partner, is an increasing function of its own overall requested quantity:*

$$\frac{d(q_i P_{q_i} - q_i P_{q_i+q_j})}{dq_j dq_i} = 0 - \frac{dq_i P_{q_i+q_j}}{dq_j dq_i} > 0 \quad \textit{\textbf{(8)}}$$

The proof for Lemma 1 as well as more detailed proofs for all following theorems, Lemmas and algorithms are available in the full version of the authors' paper[7].

Theorem 2. *If in its optimal strategy, an agent of type (q_i, n_i) chooses to engage in a search, then so does any other agent of type (q, n_i), where $q > q_i$.*

Sketch of Proof: Consider agents of types (q_i, n_i) and (q, n_i) where $q > q_i$. Since all other agents will use a reservation value strategy (according to Theorem 1) then the latter agent

[7] Can be found at www.cs.biu.ac.il/~sarned/cia/

will be accepted by all agents accepting the first. By creating a coalition with any third agent encountered by these two agents, agent (q, n_i) will gain more compared to the other type (according to Lemma 1). Therefore, since both agents search costs are identical, if the first agent preferred to search over buying directly then the second agent's strategy would be the same. []

A necessary step towards Theorem 3, which deals with the dependency of the agent's reservation vector in its quantity is the proof of Lemma 2. Here we will prove the consistency of the different elements in the reservation vectors of two agents.

Lemma 2. *For any two agent types (q_i, n_i) and (q_j, n_i), where all other agent types use reservation quantities that increase as a function of their requested quantities, the following holds: If $Q_{q_i,n_i}[k] \geq Q_{q_j,n_i}[k]$ for a specific k, then $Q_{q_i,n_i}[v] \geq Q_{q_j,n_i}[v]$ for any value v. Namely, if one agent's reservation quantity is higher than another's for any coalition size n, then this will be the case for all other coalition sizes.*

Sketch of Proof for Lemma 2: Recursively substitute the cost function of agent types (q_i, n_i) and (q_j, n_i), when using equation (7) with *n=k* for the first iteration and *n=1* for each additional step. Eventually one of the agents will reach a coalition for which it will be non-beneficial to engage in a search (according to Theorem 2, it will be agent (q_j, n_i)). This coalition price, will be the expected price per unit that agent (q_j, n_i) will pay (when adding the search costs into calculations), according to (7). The other agent's expected price per unit will be lower, since all other agent types use reservation quantities that increase as a function of their requested quantities. Repeating the same process with *n=v* will produce a contradictory result. []

Theorem 3. *A unique Nash equilibrium exists for the problem in which each agent engaging in a search uses a reservation value strategy, where the reservation value $Q_{q_i,n_i}[n]$ increases with the agent quantity q_i for every n.*

Sketch of Proof: Consider an agent of type (q_i, n_i). Assuming all other agents behave according to the theorem, we will prove that the optimal strategy for any single agent of type (q_i, n_i) is to act according to the theorem. We will use the notation: $q_n[q_i, n_i] = \max\{q \mid Q_{q,n}[n] \leq q_i\}$, the maximal quantity requested to be purchased by any of the agent types *(*,n)*, accepting agent type (q_i, n_i). $\bar{n}(q_i, n_i)$ will denote the last member of the vector $\vec{Q}_{q_i,n_i}$.

Notice that $V_{q_i,n_i}(\vec{Q}_{q_i,n_i})$ can be written as:

$$V_{q_i,n_i}(\vec{Q}_{q_i,n_i}) = E\left[a + bn_i + \frac{q_i V_{q_i+q_j,n_i+n_j}}{q_i+q_j} \bullet 1(dualaccept) + V_{q_i,n_i}(\vec{Q}_{q_i,n_i}) \bullet 1(not(dualaccept))\right] \quad \textbf{(9)}$$

Here $1(dualaccept)$ represents the indicator of the event {both agents accept each other}. If all other agents act according to the theorem, then:

$$\frac{q_i V_{q_i+q_j,n_i+n_j}}{q_i+q_j} \bullet 1(dualaccept) = q_i \sum_{n=1}^{\bar{n}(q_i,n_i)} \left(\int_{q=Q_{q_i,n_i}[n]}^{q_n[q_i,n_i]} \frac{V_{q_i+q,n_i+n}}{q_i+q} f(q,n)dq \right) \quad \textbf{(10)}$$

and:

$$V_{q_i,n_i}(\vec{Q}_{q_i,n_i}) \bullet 1(not(dualaccept)) = V_{q_i,n_i}(\vec{Q}_{q_i,n_i}) \left(1 - \sum_{n=1}^{\bar{n}(q_i,n_i)} \left(\int_{q=Q_{q_i,n_i}[n]}^{q_n[q_i,n_i]} f(q,n)dq \right) \right) \quad \textbf{(11)}$$

Substituting (10) and (11) in (9), we obtain:

$$a + bn_i = \sum_{n=1}^{\bar{n}(q_i,n_i)} \left(\int_{q=Q_{q_i,n_i}[n]}^{q_n[q_i,n_i]} \left(V_{q_i,n_i}\left(\vec{Q}_{q_i,n_i}\right) - \frac{q_i V_{q_i+q,n_i+n}}{q_i+q} \right) f(q,n)dq \right) \quad \textbf{(12)}$$

and according to (7):

$$a + bn_i = \sum_{n=1}^{\bar{n}(q_i,n_i)} \left(\int_{q=Q_{q_i,n_i}[n]}^{q_n[q_i,n_i]} \left(\frac{q_i V_{q_i+Q_{q_i,n_i}[n],n_i+n}}{q_i + Q_{q_i,n_i}[n]} - \frac{q_i V_{q_i+q,n_i+n}}{q_i+q} \right) f(q,n)dq \right) \quad \textbf{(13)}$$

Notice that $q_i + Q_{q_i,n_i}[n] > q_i + q$, for any value of q within the interval boundaries. Therefore, according to theorem (2) either at least one of the agent types $\left(q_i + Q_{q_i,n_i}[n], n_i + n\right)$ and $(q_i + q, n_i + n)$ will engage in a search or both will purchase the product in their current configuration. In the latter case, the integrated function becomes $q_i P_{q_i+Q_{q_i,n_i}[n]} - q_i P_{q_i+q}$, which is an increasing function of q_i for any value of q within the interval boundaries (according to Lemma 1), and a decreasing function of $Q_{q_i,n_i}[n]$ (due to the nature of the price function, as reflected in (1)). Also, since the other agent's reservation quantity is increasing function of q, the interval upper limit, $q_n[q_i,n_i]$, becomes an increasing function of q_i. Thus, increasing q_i must be accompanied with an increase in the interval lower limit $Q_{q_i,n_i}[n]$ of at least one of the aggregated integrals, in order to maintain the equality. Using Lemma 2, we conclude that all other elements of $\vec{Q}_{q_i,n_i}$ also increase. []

To conclude the equilibrium analysis, we suggest theorem 4, which complements theorem (2).

Theorem 4. *If in its optimal strategy, an agent of type (q_i,n_i) chooses not to engage in a search, then the same holds for any other agent of type (q,n_i) where $q < q_i$.*

Sketch of Proof: Consider an agent (q_i,n_i) for which:

$$a + bn_i = \sum_{n=1}^{\bar{n}(q_i,n_i)} \left(\int_{q=nq}^{q_n[q_i,n_i]} \left(q_i P_{q_i} - \frac{q_i V_{q_i+q,n_i+n}}{q_i+q} \right) f(q,n)dq \right) \quad \textbf{(14)}$$

Obviously this agent is indifferent to maintaining a search or purchasing the product in its current coalition structure. Notice that since the lower bound of the equation is not influenced by changes in the agent's type, then according to Lemma 1, the right hand side of the equation is an increasing function of q_i. Therefore, there is a specific agent type (q_i,n_i), where all other agent types (q_j,n_i), where $q_j < q_i$, will never engage in a search. []

Theorems 1-4, fully outline the conditions by which an agent will decide to engage in a search in order to extend its coalition, and the characteristics of its optimal strategy within the search. An important issue to consider is the complexity of the equations from which the equilibrium strategies can be extracted. The different characteristics of the optimal strategies as described in this section can aid us in this task. We will demonstrate such a process using a variant of the problem in the following section.

4. Specific Cases

In this section we discuss some variants of the general coalition formation problem modeled in the previous section. The first variant, in which the coalition coordination costs are negligible, is an upper limit for the agent types that will engage in a search in the general model. The second variant, the two-size coalition is a good example and a test bed for demonstrating possible uses of algorithms for distributed calculation of the agent's optimal strategy parameters in equilibrium.

4.1 Coordination costs are negligible

Consider a marketplace where the agents are not subject to coalition coordination costs, *bn*, and the only cost associated with the search is the fixed cost per search stage, *a*. Here, there is no significance to the number of buyers a potential partner agent is representing, and the only relevant decision argument is the partner's overall requested quantity. As long as the agent engages in search it will accept any agent it encounters (the search cost is a sunk cost, and any coalition will increase the coalition quantity).

An agent of type $(q_i,*)$ will choose to search, only if the expected coalition to be reached through the search will benefit more than buying the product in the current coalition configuration. Denoting the overall cost of agent type $(q_i,*)$ if it is conducting a search as $V_{q_i}(search)$, and the cost of purchasing the product without a search as $V_{q_i}(buy)$, we obtain:

$$V_{q_i}(search) = a + E\left[\frac{q_i V_{q_i+q}}{q_i + q}\right],\ V_{q_i}(buy) = q_i P_{q_i} \qquad \textbf{(15)}$$

Remember that $f(q,n)$ is the distribution function of the agent types engaging in a search, in equilibrium, therefore:

$$E\left[\frac{q_i V_{q_i+q}}{q_i + q}\right] = \int_{q=\underline{q}}^{q_{upper}} \frac{q_i V_{q_i+q}}{q_{i+}q}\left(\sum_n f(q,n)\right)dq \qquad \textbf{(16)}$$

In the above equation (16), q_{upper} represents the highest quantity for which *f(q,*)>0*. Agent $(q_i,*)$ will prefer to conduct a search if the following equation (17) holds, and otherwise it will prefer to buy the product in its current configuration:

$$V_{q_i}(buy) - V_{q_i}(search) = q_i P_{q_i} - a - \int_{q=\underline{q}}^{q_{upper}} \frac{q_i V_{q_i+q}}{q_{i+}q}\left(\sum_n f(q,n)\right)dq > 0 \qquad \textbf{(17)}$$

Notice that in equilibrium all agents of type *q,* where $q > q_{upper} - \underline{q}$, will not form further coalitions and though the term $(q_i V_{q_{i+}q})/(q_{i+}q)$ can be expressed as $\overline{q}_i P_{q_{i+}q}$.

4.2 Agent pairs' coalitions

In this section we consider a second variant of the problem, concerning coalitions of agent pairs. Each coalition can be seen as a beneficial partnership among two autonomous agents, of specific types *(q,1)*. The reason for limiting the discussion here to coalitions of pairs is twofold. First, the dynamics of the atom (size-two) coalition structures creation processes is a basic infrastructure for the analysis of the larger coalitions. Thus this can be a useful test bed for algorithms that can be extended later to deal with the general case. Second, we identify several applications where an agent can benefit or perform a task only by finding one partner[8].

Consider, for example, a client agent in a Kazaa\Gnutella-like file sharing application, searching for a specific media file A, and offering another specific media file B. The agent can find a partner with complimentary offerings, starting immediately with the download and upload processes. Alternatively, it can look for a better connected agent with the same offering, in order to reduce the download time. At each stage of the search, the agent will have to consider the tradeoff between time saved (on download, by possibly finding a better partner) and time spent (in search for such a partner). In addition, when considering possible better partners, the agent should take into account these agents' own strategies and asses their willingness to form a coalition with its type. A totally different application can

[8] We do understand the difficulties of using the abstract size-two model in these applications. However the analysis given, as well as the algorithms to follow, are unique in the context of coalition formation for these applications.

be found in VoIP networking. Here, we can utilize the partnership concept, for service providers, looking for partners to form ad-hoc call terminations between two destinations. Each service provider faces different partners offering different link qualities (jitter, packet loss, etc.). However testing a partner's quality also entails costs.

In the context of the general coalition formation problem in electronic marketplace, as described in section 3, we may consider this specific variant when there might be technical implementation obstacles for creating a representative agent for more than one buyer. Though, for an agent that has decided to engage in a search, at each stage of its search process, after reviewing the information regarding the current potential partner, the agent must make a decision whether to terminate the search or to continue. Terminating the search will result in immediate purchase of the product with the current potential partner, if it agrees, whereas continuing the search will hopefully yield a better type partner.

Notice that since the agent is not concerned with coalition coordination costs, the only relevant parameter for its decision is the other agent's requested quantity. Therefore the agent will be using a reservation quantity Q, instead of the reservation vector $\vec{Q}$ as in the general case. When a mutual acceptance occurs the agent ends its search and purchases the product at price $P_{q_i+q_j}$. Equation (9) can now be expressed as:

$$V_{q_i}(Q) = E\left[k + q_i P_{q_i+q_j} \bullet 1(dualaccept) + V_{q_i}(Q) \bullet 1(not(dualaccept))\right] \qquad \textbf{(18)}$$

where $V_{q_i}(Q) = V_{q_i,1}(Q[1])$ and $k=a+b$.

Theorem 5. *For this variant of the problem a unique Nash equilibrium exists for the agent types engaging in a search, in which each agent uses a reservation value (reservation quantity) strategy, with a lower reservation than its requested quantity (derived by its type), and increases with the agent's type*[9].

Sketch of Proof: The proof methodology resembles the one given for theorem (3). The equivalent to Equation (11) is:

$$V_{q_i}(Q_i) = E\left[k + q_i P_{q_i+q_j} \bullet 1(Q_i \le q \le q(q_i)) + V_{q_i}(Q_i) \bullet 1(q_j < Q_i \cup q_i < Q_j)\right] \qquad \textbf{(19)}$$

And using $V_{q_i}(Q_i) = q_i P_{q_i+Q_i}$ (the equivalent to equation (13) of the general case), we obtain:

$$k = q_i \int_{q_j=Q_i}^{q(q_i)} \left(P_{q_i+Q_i} - P_{q_i+q_j}\right) f(q)dq \qquad \textbf{(20)}$$

Concluding that the only way to maintain the equality, when increasing q_i is by increasing Q_i. Since all buying agents will accept agent type $\overline{q}$, then this agent's type problem becomes a simple problem of choosing the best reservation quantity without any restriction (of acceptance by other agents). This leads to $\overline{Q} < \overline{q}$, and so therefore $Q_i < q_i$ for any q_i. []

As in the general problem, if an agent requesting a quantity q_i finds it beneficial to engage in a search, rather than buy the product as a single buyer, then any other agent requesting a quantity $q > q_i$ will find the search process beneficial as well. Consider the agent type q_j, where $V_{q_j} = P_{q_i} q_i$. Since for all agents of type i above q_i, $V_{q_i} = P_{q_i+Q} q_i < P_{q_i} q_i$ holds, then these agents have no incentive to abandon the search. Agents of a type lower than q_j, will prefer to leave the market and not to engage in a search at all. The q_j type can be calculated in the continuous case as follows:

$$k = q_i \int_{q_j=\underline{q}}^{q(q_i)} \left(P_{q_i} - P_{q_i+q_j}\right) f(q)dq \qquad \textbf{(21)}$$

[9] Similar characteristics of equilibrium as in theorem 5 are described in [9] for a model concerning the maximization of the agent's utility when searching for pairs.

Once agents of a specific type have an incentive to leave the environment, new equilibrium reservation quantities should be computed for the remaining agents. The new calculation should be based on the updated probability function of the remaining agents. This procedure should be repeated until all agents have an incentive to buy through pair coalitions.

Consider equation (20). We have shown that the right hand side of the equation is an increasing function of Q_i, and though once $q(q_i)$ is known, Q_i can be calculated by setting $Q_i = q_i$, and decreasing this value until the right hand side exceeds the value *k*. Notice that all the agents will accept the highest type agent, $\overline{q}$, since an agent's reservation quantity is always smaller than its requested quantity. This means that agents of type $\overline{q}$ can simply calculate their reservation quantity $\overline{Q}$ by solving (20) for $q(\overline{q}) = \overline{q}$, regardless of the other agents' reservation quantities. This calculation method is valid also for all agents in the perceived interval $[\overline{Q}, \overline{q}]$, since all other agents' reservation quantities are lower than $\overline{Q}$. Furthermore, for each one of the agent types in this interval, a unique finite sequence of reservation values can be obtained. Each member of such sequence (except for the first one) is the reservation quantity of the former sequence member. Each sequence member (except for the last one) is also the highest type agent willing to accept agents of the next sequence member type. Thus, each sequence member can be calculated using (20), with the former sequence member agent as the integral upper bound.

Obviously, calculating all possible sequences will reveal the reservation values for all agent types in the interval $[\underline{q}, \overline{q}]$. However, having a continuous range of agent types makes the task impossible. A computational algorithm is needed to find a reservation quantity of a given agent of type q_i. Utilizing the unique characteristics of the equilibrium, along with the concept of the sequences, we suggest an algorithm for bounding the optimal reservation quantity, Q_i, into an interval of size ε.

Recall that each agent's reservation quantity is smaller than its requested quantity, and increases as its type increase. Therefore Q_i is bounded by the reservation quantities of two subsequent agents in any sequence. For the purpose of bounding Q_i in an interval of size ε, we should find two agent types q_{i_-} and q_{i_+}, whose reservation quantities difference is smaller than ε. This is done by changing the selection of the two sequence originating types in the interval $[\overline{Q}, \overline{q}]$. Using binary search over this interval will ensure that on each step the interval bounding Q_i is narrowed. The proposed algorithm for computing such a sequence, which is valid for any distribution function $f(q)$, and search price k, is specified below.

Algorithm 4.1 (FindSequence(q_{start}, q_{stop})).

(* Input: (1) An agent type q_{start} to start with ; (2) An agent type q_{stop} to stop with *)
(* Output: a sequence, represented by an array, q[], where q[0]= q_{start} , and q[last] is the first sequence member exceeding type q_{stop} from below *)

```
1. q[0]:= q_start;
2. q[1]:=Q_i, where Q_i solves (20) with q_i=q[0] and q(q_i)=q[0];
3. i:=1;
4. While (q[i] ≥ q_stop) do {i++; q[i]:=Q_i, where Q_i solves (20) with
   q_i=q[i-1] and q(q_i)=q[i-2]; }
5. return q[];
```

Using the above procedure, the following scheme can be used to calculate the reservation quantity of agent type q_i:

Algorithm 4.1 (CalculateReservation(q_i, ε)).

(* Input: (1) An agent type q_i ; (2) level of precision ε *)
(* Output: the bounded interval for reservation quantity of type q_i *)

```
1.  q_upper:=FindSequence (q̄, q_i);
2.  If (q_i ≥ q_upper[1]) then
     a. Upper=Q_i, where Q_i solves (20) with q_i and q(q_i):=q̄;
     b. Return (Upper, Upper);
3.  q_lower= FindSequence (q_upper[1], q_upper[last]);
4.  While (q_upper[last]- q_lower[last]>ε) do
     a. q:=FindSequence((q_upper[0]+q_lower[0])/2, q_upper[last]);
     b. if (q[last]>q_i) then q_upper=q;
     c. else q_lower=q;
5.  return(q_upper[last]+q_lower[last]);
```

Further explanations and proofs regarding the correctness of the algorithms are available in the full version of the authors' paper.

Notice that in the absence of the algorithm, any given agent would have been required to compute reservation values for a continuous interval of types. The proposed algorithm suggests evaluation of the reservation value, in a finite number of steps, for any level of required precision.

4.3 Discrete environment

The above algorithms are concerned with markets characterized by enumerated types like gold, coffee and chemicals. However for most products in the electronic marketplace, the required quantity can be expressed only in discrete units. In the following paragraphs, we will demonstrate how these algorithms can be adjusted to handle such discrete environments. This can be obtained by proving that the same equilibrium characteristics found for the continuous case are also maintained in the discrete environment.

Theorem 6. *Consider a finite set of n agent types, each characterized by (q_i, g_i), where q_i is the quantity requested by agent type i, and g_i is the proportion of this agent type in the population. Then there is a unique Nash equilibrium for the model, where agents use reservation values as the optimal strategy, and:*

(a) An agent's reservation quantity is an increasing function of its type.

(b) An agent's reservation quantity is lower than its requested quantity.

(c) Type q_i agent's reservation quantity Q_i, satisfies:

$$k = q_i \sum_{q_j=Q_i}^{q(q_i)} \left(P_{q_i+Q_i} - P_{q_i+q_j}\right) g_j \qquad \textit{(22)}$$

Sketch of Proof: Similar proof to the one given in section 4.2, resulting in equation (22). Using Lemma 1 (the Lemma concerns only the price function and its validness is not influenced by the distribution of types), we conclude that in order to keep the above equation valid, Q_i must be an increasing function of q_i. []

As in the continuous case, from lemma 1 we conclude that the right hand side of (22) is an increasing function of Q_i, and though once $q(q_i)$ is known, Q_i can be calculated. Therefore, a similar algorithm to that presented in section 3.2 for finding a reservation quantity of a given agent of type q_i, can be applied in the discrete case, replacing equation (20) with (22). Notice, however, that in the discrete case there is a possibility that calculations based on (22) will result in a reservation quantity Q_i, with no actual agent type *j* associated to it. Here, any reservation quantity taken from the interval $[Q_i, \min(q_j); q_j \geq Q_i]$ will result in identical expected minimal costs. Therefore, for each agent we can define a reservation type instead of a reservation quantity. The reservation type defines the lowest agent type an agent will be willing to accept as a partner. An appropriate reservation agent value for an agent *i* type can be calculated simply by using equation (22): first, find the upper index based on the reservation quantities of higher types (starting with the highest type agent and backward). Then, check the right hand value when setting $Q_i = q_j; j = i, i-1, \ldots$, until the calculated term exceeds the value k. This will require overall (n-j+1) reservation type calculations. Notice that once the reservation type concept has been adopted, an agent's reservation type is a weakly increasing function of the agent's type (several agents may have the same reservation agent). In addition, the agent's reservation type can actually be its own type (and not necessarily a lower type).

For the purpose of calculating Q_i, we can utilize the new acceptance types concept, suggesting a more efficient heuristic. The algorithm works on the array agents[] where each member *i*, agent[i], holds, in addition to the requested quantity, and type's probability fields, also pointers to the highest agent type that accepts type *i*, and the lowest type accepted by agent *i*.

The algorithm uses two functions:

- FindUpper(i,initial) - used to associate a given agent of type *i* with the highest agent type that will accept it. This value will be used as $q(q_i)$ in the calculation of Q_i in (22). The function is initialized with a higher or equal agent type, *initial*, which is known to accept type *i*.
- FindSequence(q_{start}, q_{stop}) – A similar function as in the continuous case. This time, however, each member is the lowest agent type accepted by the former sequence member. The last member in the array is the first sequence member smaller than type q_{stop}.

***Algorithm 4.2* (FindUpper(i,initial)).**

(* Input: (1) Agent type i for which the calculation is requested *)
(* (2) An initial agent type to start with *)
(* Output: Updating agents[i].upper with appropriate value *)

```
1. If (agents[i].upper==null)
   then {
2. j=initial;
3. While (j<n) do {
4. i=i+1;
5. If(CalculateReservation(i)>i)
   agents[i].upper=j-1;
   }
6. If (i=n) then
   agents[i].upper:=n;
}
```

***Algorithm 4.2* (FindSequence(q_{start}, q_{stop})).**

(* Input: (1) An agent type q_{start} to start with *)
(* (2) An agent type q_{stop} to stop with *)
(* Output: array, q[], representing a sequence, where q[0]= *)
(* q_{start}, and q[last]= first agent type smaller than *)
(* q_{stop} belonging to such a sequence *)

1. q[0]:= q_{start} ; FindUpper(q[0],q[0]);
2. q[1]:= CalculateReservation(q[0]);
3. i:=1;
4. While (($q[i] \geq q_{stop}$) and (q[i]!=q[i-1])) do{
   a. i++;
   b. FindUpper(q[i-1],q[i-2]);
   c. q[i]:= Q_i , where Q_i solves (22) with q_i =q[i-1] and $q(q_i)$ =q[i-2];}
5. return q[];

***Algorithm 4.2* (CalculateReservation(i)).**

(* Input: (1) An agent type i for which the reservation type should be calculated *)
(* Output: Returns agent type i's exact reservation type *)

1. Base cases: If i is highest or lowest type agent, or upper index was already calculated then return(Calculate Q_i by Solving (22));
2. Heuristic: if i-1 type is known then FindUpper(i,agents[i-1].upper) and return(Calculate Q_i by Solving (22));
3. Use findSequence(n,i) to set q_{upper} and q_{lower} as the two initial bounding arrays as in section 4.2.
4. First Segment: Set Agents[j].upper=n for every $j \geq q_{upper}[1]$;
5. Repeat
   a. *i* is in first segment: If ($q_{lower}[0] < i$) then set agents[i].upper= $q_{lower}[i]$ and return(Calculate Q_i by Solving (22));
   b. Both sequences collide: If $q_{lower}[j] = q_{upper}[j]$ for any $j \leq length(q_{lower})$, then:
      i. Shorten both arrays so that q_{upper} will start with its member j, and q_{lower} with its member j+1;

ii. For any $q_{lower}[j] \le k < q_{lower}[j]$, FindUpper($k, q_{upper}[i]$);
c. Adjacent members of sequences: If $(q_{upper}[j] - q_{lower}[j]) = 1$ then:
i. for any $q_{lower}[j+1] < k < q_{upper}[j+1]$ Set agents[k].upper=$q_{lower}[j]$;
ii. Shorten both arrays so q_{upper} and q_{lower} will start with their j+1 member;
d. *i* is in new first segment: If ($(q_{lower}[0]) \le i$) then
i. FindUpper($i, q_{upper}[0]$);
ii. Return (Calculate Q_i by Solving (22));
e. $q = findSequence((q_{upper}[0] + q_{lower}[0])/2), q_{upper}[last]$;
f. Update bounding sequences: If (q[last]=i) then
i. FindUpper(q[last],q[last-1]);
ii. Return (Calculate Q_i by Solving (22));
g. Else if (q[last]>i) then $q_{upper} = q$;
h. Else $q_{lower} = q$;

The function *CalculateReservation* returns the lowest agent type an agent *i* will accept in its equilibrium optimal strategy.

In the worst case, where each agent type's reservation quantity equals its own quantity, the algorithm will calculate a reservation quantity for all agents with a type higher than *i* (since we are considering discrete types and Q_i can be easily calculated using (22)). Using simulations, we tested the proposed algorithm performances when creating random agent types with random probabilities. For each simulation we set a random search cost, and used a typical price function[10] (according to (1) and (2)). Agent type distribution was created randomly for each simulation, and the discrete types (quantities) were randomly drawn for each simulation from the interval (1,10). We summarize the results of the simulations in Figure 3.

Figure 3 describes the average[11] extent of usage of equation (22), which is the most resource intensive calculation module in any algorithm within this problem's context. The horizontal axis represents the number of agent types used. The vertical axis represents the percentage of agent types for which we had to calculate a reservation quantity, out of the total number of agents with a higher type than the required one. In the absence of a heuristic, an agent is forced to calculate all reservation quantities for types greater than its own (represented as the one hundred percent line in the graph), thus the time required by our proposed heuristic is significantly lower than any straightforward algorithm. Notice that the algorithms performance improves as the environment (number of agent types) increases.

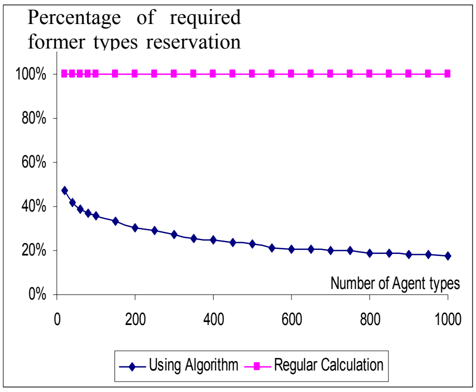


Figure 3 – Algorithms performance

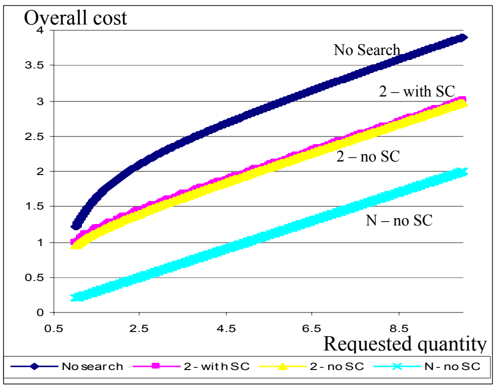


Figure 4 – Utility Comparison

[10] We used a price function of type a-b*ln(c*X).
[11] Each graph point is the average of 5000 random simulations for calculating a reservation agent for a type outside the first (trivial) segment.

Finally, we present in Figure 4, an example of the agent's utility graph, in the context of a search (Recall that the lower the overall cost, the better the agent's utility). Four scenarios were analyzed using direct calculation and simulation, where the agent types were drawn randomly from the interval [1,10]. Search cost was ~0.5 cents, and product cost 0.2+2/x-1/x^2, where x is the required quantity. In the first scenario (represented by the most upper curve, marked as *no search* in the legend), agents did not engage in a search and purchased the product with a cost correlated to their required quantity. Secondly, we used the simulation to obtain the agent's costs when conducting a search in a costly environment (represented by the middle upper curve, marked as *2 – with Search Costs* in the legend). The reservation quantities were calculated for each agent according to the proposed algorithms. Then, an upper limit for the search benefits was set, by calculating the agent's cost when searching for partners in a non-costly environment (represented by the middle lower curve, marked as *2 – no Search Costs* in the legend). This is not the case of one united coalition but rather a solution where each agent forms a coalition with another agent of its own type[12] (a known result for the two-size costless model – see [1]).

A theoretical n-size coalition with no search costs is represented by the lower curve (marked as *N – no Search Costs*). Here all agents form together one big coalition, and buy the product with a price $P_{q\to\infty}$. This curve can be seen as a lower bound for cost of coalitions in the n-size scenario with search costs.

In Figure 4, all agents have an incentive to conduct a search. The case where agent's cost when buying directly the product is higher than when conducting a search, is well supported by the model, and this agent type will abandon the search in equilibrium. This was fully analyzed in section 3. Therefore in equilibrium, all agents engage in search will benefit more than the cost of buying directly the product.

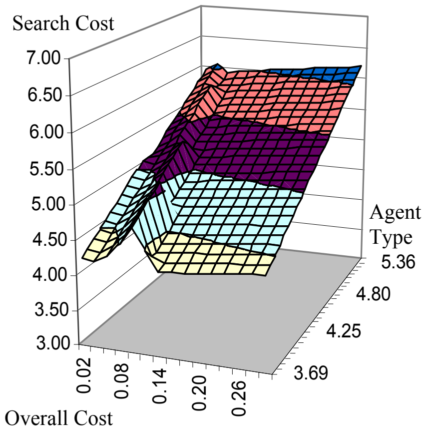


Figure 5 – Agent type and search cost effects

Notice that in figure 4 the difference between the results obtained from t

he costly search and those belonging to the theoretical costless environment are significantly small. However, this is well correlated with search cost. An example of the effect the search cost has on the perceived utility for a given agent type is given in figure 5. As the overall search costs increase, two contrast effects occur: the agent becomes more attractive to higher type agents since the benefit of being selective decreases, and the agent's own payments for search increases. These two factors are the reason for the local maximum in figure 5.

5. Conclusions

In this paper we analyzed the model and equilibrium optimal strategies for agents engaged in costly searches for potential coalition partners. The incentive for cooperation is

[12] As if all buyers get together in one place and find pairs that belongs to the core.

a volume discount and the search is characterized with a fixed cost for locating partners and variable costs for sub-coalition coordination towards the final coalition structures.

Understanding the characteristics of agent strategies in equilibrium is the main brick stone for any algorithm and heuristic to be used for solving the general case. We have shown a comprehensive analysis of two algorithms for solving specific variants of the problem where the coalition size is restricted to pair partnerships. We do see these algorithms as a basic infrastructure for suggesting further algorithms for the general n-size case. This is mainly because of the similar equilibrium structure (the continuum of agents engaged in a search) and the special characteristics of the agent's strategies as we proved in this paper (reservation vectors\values increase in type). The proposed tools for calculating the equilibrium strategies of agents when searching in pair coalitions can be used in several plausible applications and environments (Gnutella\Kazaa, VoIP service providers, etc.).

We plan to extend the research towards completing the algorithms to be used in the general case and simulations that can describe the evolution of steady-state agent types' distributions. Though we have focused on the non-transferable utility case, we see great importance in understanding the changes in such models when the agents can negotiate over the surplus of the partnership.

References

1. G. S. Becker. Theory of marriage: Part I, *Journal of Political Economy*, 81(4):813-846, 1973.
2. S. Breban, J. Vassileva. Long-term Coalitions for the Electronic Marketplace, *in B. Spencer (ed.) Proceedings of the E-Commerce Applications Workshop, Canadian AI Conference*, Ottawa, 2001.
3. C. H. Brooks, E.H. Durfee and A. Armstrong. An Introduction to Congregating in MultiAgent Systems. *In Proc. of ICMAS*, Boston, 2000.
4. K. Burdett and R. Wright. Two-sided search with nontransferable utility, *Review of Economic Dynamics*, 1:220–245, 1998.
5. A. R. Greenwald and J. O. Kephart. Shopbots and pricebots. *In Proc. of IJCAI*, 1999.
6. K. Lermann, O. Shehory. Coalition Formation for Large Scale Electronic Markets. *Proc. of ICMAS*. 2000.
7. C. Li and K. P. Sycara. Algorithm for combinatorial coalition formation and payoff division in an electronic marketplace. *AAMAS 2002*: 120-127.
8. J. McMillan and M. Rothschild. Search, *Ch.27 of Handbook of Game Theory with Economic Applications,* Vol. 2, 905-927, ed. Robert J. Aumann and Sergiu Hart, Amsterdam, 1994.
9. P. Morgan. A Model of Search, Coordination, and Market Segmentation, *mimeo*, SUNY Buffalo, 1995.
10. T. Sandholm, K. Larson, M. Andersson, O.Shehory, F. Tohme. Coalition structure generation with worst-case guarantees. *Artificial Intelligence*. 111:209—238, 1999.
11. T. Sandholm and V. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94:99-137, 1997.
12. S. Sen and P.S. Dutta. Searching for optimal coalition structures. *In Proceedings of ICMAS*, 2000.
13. O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. *Artificial Intelligence*, 101:165-200, 1998.
14. L. Smith. The marriage model with search frictions. *Journal of Political Economy*, 2003 (To appear).
15. N. Tsvetovat, K. P. Sycara, Y. Chen and J. Ying. Customer Coalitions in Electronic Markets, *AMEC*, 2000.
16. J. Yamamoto and K. Sycara. A Stable and Efficient Buyer Coalition Formation Scheme for E-marketplaces. *Proceedings of Autonomous Agents*, 2001.