ICEB 2004 Proceedings

International Conference on Electronic Business (ICEB)

Winter 12-5-2004

A Framework to Support Coalition Formation in Supply Chain Collaboration

Chattrakul Sombattheera

Aditya K. Ghose

Peter Hyland

A Framework to Support Coalition Formation in Supply Chain Collaboration

Chattrakul Sombattheera, Aditya Ghose, Peter Hyland

Decision Systems Laboratory, School of IT and Computer Science,
University of Wollongong, NSW 2500, Australia
{cs50, aditya, phyland} @uow.edu.au

ABSTRACT

This paper proposes a framework for agents in globally collaborative supply chain by applying the concept of coalition formation, a cooperation game in game theory. This framework provides mutual benefits to every party involved buyers, sellers and logistics providers. It provides a common gateway that allows individual parties to locate the right partners, negotiate with them, and form coalition in the best possible ways. The framework is applicable to real world e-business models, including B2C, B2B, supply chain and logistics, SME, etc. We firstly discuss common needs existing in today e-business. We then discuss about our framework, i.e., negotiation protocol and decision mechanism.

Keywords: coalition formation, game theory, negotiation protocol, decision mechanism

1. INTRODUCTION

The current business environment is characterized by large complex supply chains that are often global in reach and that are highly adaptive, being frequently re-configured to respond to dynamic business contexts. It is widely recognized collaboration across the supply chain is a key prerequisite for supply chain efficiency. The effective deployment of information technology and systems, specially the Internet and the worldwide web, has made new modes of complex, dynamic yet effective collaboration possible across supply chains. Collaboration in supply chains can take various forms. Suppliers and buyers might collaborate to increase buying/selling power, to reduce logistics costs or to aggregate capacity. The problem of determining an optimal set of collaborative arrangements/agreements for a given firm can be complex. This research seeks to develop automated (or semi-automated) negotiation protocols as a basis for building decision support functionality for dealing with this complexity.

The coalition formation problem ([4], [9]) considers techniques and criteria that might be used by a collection of (rational) agents to decide how they might group together to improve individual or social utility. Coalitions are ubiquitous in real-life settings. The theoretical underpinnings of approaches to coalition formation lie in the literature on multi-player games in game theory ([9]). Players negotiate among themselves about payoffs to decide which coalition to join. In reality, it is more complex than that. Self-interested agents, operating in dynamic environment such as supply chain, are under heterogeneous constraints. Furthermore, each agent has its own strategies that increase or decrease the value of each constraint thus affects the decision making of agents. A simple example of this is one is using express mail service, which costs more, due to a deadline while another is using ordinary mail service, which costs less, with further deadline. While time is changing in dynamic environment, the value of a constraint varies thus affecting the utility of the agent. In contrast to traditional coalition formation study, where the coalition value is predefined and thoroughly known among agents, an agent has to calculate, according to its constraints and strategies, for the coalition that would give it maximum utility. We believe that two of the key components of successful coalition formation of self-interested agents are, one, quickly negotiating with other agents and, two, selecting the best possible coalition. Each agent, bounded by its own constraints, may negotiate with others to form a coalition, which is likely to yield maximum benefit. Such a coalition, however, may not be formed due to the constraints. So the agent has to look for the next best possible coalition by consulting with its internal utility mechanism. Negotiation and decision must be done in a timely fashion. There are works in coalition formation that discuss the formation of buyers, sellers and LPs separately. This work provides a basic framework that allows thorough collaboration among agents in supply chain. It includes two important components: a negotiation protocol and a decision mechanism. The negotiation protocol allows agents to exchange necessary information before deciding which coalition to join. A coalition can be as complex as a coalition of buyers, sellers and LPs. **We restrict attention to traditional supply chain activities for the sake of simplicity. Our focus is primarily on material flows, but note that many of the concepts developed in this work could apply equally well to these other forms of collaboration.**

2. COLLABORATION IN THE SUPPLY CHAIN

As discussed earlier, our focus in this paper is to develop coalition formation techniques that are appropriate to large, complex supply chains. In our domain, we can conceive of three categories of actors/agents: *buyers*, *sellers* and *logistics providers (LPs)*. In this view, a logistics provider is any organization that provides transportation, warehousing or other logistics-related services. A firm (buyer or seller) has a set of plans to produce goods. Each plan consists of multiple activities, which may be of two types: *externa*l and *internal*. External activities are those that involve with *i*) procuring raw materials from external suppliers and *ii*) transporting the raw materials from the suppliers' warehouses to appropriate manufacturing sites. Internal activities are those that deploy internal resources of the firms (e.g., in-house manufacturing, assembly etc.). These activities are often required to satisfy synchronization/scheduling constraints, and there are penalties associated with the violation of these constraints. In addition to these constraints, each agent/actor seeks to satisfy several (internal) objectives, e.g. buying the raw materials at the affordable prices, with acceptable quality etc. In modern web-enabled supply chains, the process of matching buyers with sellers (of goods or services) is often facilitated by online e-marketplaces. Our framework assumes the existence of multiple such e-markets, where sellers advertise their product or service offerings while buyers advertise their requirements. There are many different motivating factors that make firms collaborate. In other words, there are multiple distinct drivers for coalition formation.

The simplest form of a supply chain coalition is one where a buyer forms a coalition with a seller and an LP to satisfy one of the external activities in a plan (e.g. to supply and deliver a manufacturing input to a production process). Buyers sometimes form coalitions with other buyers for the purpose of buy-side aggregation, i.e., the aggregation of buying power. Sellers sometimes form coalitions with other sellers to aggregate selling power. Such coalitions are common for small sellers, such as in agricultural cooperatives (e.g. for micro-producers of dairy products). Logistics provider (LPs) may want to form coalitions to aggregate their service capacities. We note that we have only listed the basic drivers. Most real-life supply chain coalitions tend to have more than one motivating factor driving their formation. Thus a coalition consisting of several sellers and several LPs might be driven by the need to aggregate selling power (for the sellers), the need to aggregate production capabilities (again for the sellers) and the need to offer comprehensive production and delivery contracts (thus bringing in one or more LPs into the coalition). We also note agents/actors (i.e., buyers, sellers or LPs) could participate in more than one coalition at any given point in time. Thus a seller might participate in a coalition with other sellers to aggregate selling power, while simultaneously participating in a coalition with a different set of sellers in order aggregate manufacturing capabilities.

Both quantitative and qualitative factors play a role in coalition formation decisions. Price, delivery cost and lead time are examples of the former while quality, satisfaction and trust are examples of the latter. In the current work, we focus on four key criteria - cost, lead time, quality and trust - while acknowledging that other criteria might be relevant (though possibly less important). Cost is the sum of price and delivery cost. Lead time is the time elapsed between an order being placed and the good or service becoming available. Trust is usually taken to denote the degree of belief of an actor/agent that another actor/agent will live up to its commitments. These attributes are not orthogonal, and most decisions typically involve trade-offs between them. One may be willing to pay more for a shorter lead time (or higher quality or higher trust/reliability) and vice versa. One may, in some settings, be willing to accept longer lead times for the purpose of getting higher quality. Several other instances of such trade-off are common, but we do not list them all here. Although negotiation can be either bilateral or multilateral, we can in most instances reduce negotiation to the bilateral case without loss of generality. The simplest scenario is where single agents negotiate, e.g., a buyer with a seller and a buyer with an LP. A buyer can bilaterally negotiate with multiple sellers/LPs at the same time in order to find the best pair of a seller and an LP. The information exchanged is kept private to each agent. A more complex scenario is where multiple agents negotiate. An agent that stands to benefit the most from forming a coalition usually acts as the coalition leader. A coalition leader negotiates bilaterally with multiple agents of its own type (buyer, seller or LP) to establish a sectoral coalition, i.e., a coalition of buyers, or a coalition of sellers, or LPs etc. The object of such negotiation is usually to establish how the coalition value, i.e., the financial benefits of collaboration might be distributed across the members of the coalition.

3. A FRAMEWORK FOR SUPPLY CHAIN COALITION FORMATION

In our model, the agents are divided in three groups—the set B = $\{B_1, B_2, \ldots, B_i\}$ of buyers, the set S = $\{S_1, S_2, \ldots, S_j\}$ of sellers, and the set L = $\{L_1, L_2, \ldots, L_k\}$ of LPs. The set LOC = $\{loc_1, loc_2, \ldots, loc_m\}$ is a set of locations where sellers' manufacturing facilities or warehouses and buyers' sites are located. The set G = $\{G_1, G_2, \ldots, G_n\}$ is a set of goods. Each good g is associated with a load belonging to the set LD = $\{LD_1, LD_2, \ldots, LD_o\}$, where a load is defined by either weight or volume. The notion of a load as distinct from a good becomes relevant in the context of negotiation with LPs. The sets LOC, G and LD are common knowledge to all agents. We will assume that each agent (whether a buyer, seller or LP) has access to the following: *1*) A history of the prior coalitions that the agent has participated in. *2*) A history of what has transpired in the current round of negotiation. *3*) An *operations management system.* In the instance of buyers and sellers, these would essentially be some form of production scheduling and optimization system. For LPs, these would be some form of logistics optimization system. The intent is to be able to query the system to obtain quotes that would be used in the process of negotiation. Thus a seller should be able to query the operations management system by providing the product requirements and the date/time by which the product must be ready (typically a shipping deadline) to obtain a price quote. This quoted price is typically higher for early shipping deadlines (i.e. rush orders) and vice versa. Similarly, an LP should be able to query such a system with details on the loads, origin, destination and delivery deadlines to obtain a price quote to use in negotiation. Once again, rush orders are likely to result in higher price quotes and vice versa. An operations management system is assumed to have access to real-time data about the firm's current business context, and thus provides highly context-sensitive output. We do not discuss the design and implementation of such systems, but note that most firms (especially larger ones) tend to employ some version of these systems.

In the following, we present a simple negotiation protocol to support coalition formation. We first describe some basic assumptions made by the protocol. A message containing fields set to null is treated as a query. An agent replies to a query with an offer made by filling the null fields of the query with proposed values. Agents negotiate by modifying the values of the received offers and sending them back. Agents accept offers by sending acknowledge messages. When a deal is done, agents are obliged to discharge their commitments.

3.1 Buyer to Seller and Buyer to LP Negotiation

Direction	Message
B®S	*B2S(msg#, buyer, good, totalquantity, quality, (availabletime, origin, quantity), price, expirytime)*
B¬S	*S2B(msg#, seller, good, quality, (availabletime, origin, quantity), price, expirytime)*
B®S	*B2Sack(msg#)*
B¬S	*S2Back(msg#)*
B®L	*B2L(msg#, buyer, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)*
B¬L	*L2B(msg#, LP, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)*
B®L	*B2Lack(msg#)*
B¬L	*L2Back(msg#)*
Note: *B* is the buyer, *S* is the seller and *L* is the LP.	

Table 1: Buyers' negotiation messages with sellers over buying goods and with LP over delivering goods.

The buyer begins negotiation by sending a message *B2S*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*) to sellers asking for a quote for *totalquantity* units of the *good*. (*availabletime*, *origin*, *quantity*) is a list of triples, each of which specifies that a certain *quantity* of the good needs to be picked up from *origin* at *availabletime*. If the buyer is contacting the seller for the first time or wants an updated quote, *availabletime*, *origin*, *quantity* and *price* are set to null (i.e., the buyer does not wish to pre-specify an available-time, the exact location the good is to be sourced from, the quantity to be sourced from that location and the price the seller may wish to offer). In subsequent messages from the buyer to the seller, the (*availabletime*, *origin*, *quantity*) triples may have non-null values. This is because the seller would have replied to the buyer with a similar list of tuples (see below) indicating that the seller would be able to meet the order by sourcing production from multiple locations with distinct available times. The *expirytime* provides the seller with a deadline within which it must reply – otherwise the request will be deemed to have expired. Upon receiving the message, each seller consults its operations management system, retrieves the price quote and available time, then replies to the buyer with the message *S2B*(*msg#*, *seller*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*), indicating that the *seller* will be able to offer the *good*, at a specific *price*, ready at *origin* on *availabletime*. The offer lasts for *expirytime* (in case the seller is not in a position to supply the *good*, it will not reply). If the buyer is satisfied with the offer, it will send the message *B2Sack*(*msg#*) back to the seller. Once the seller receives it, the deal is done—the buyer is obliged to pay money and the seller has to supply the good. If the buyer is not satisfied with the price, it negotiates on the price by sending the message *B2S*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*) with the new price back to the seller again. If the seller is satisfied with it, it will send the message *S2BAck(msg#)* back to buyer in order to seal the deal.

At the same time, the buyer also contacts one or more LPs to organize shipping of the product. It sends the message *B2L*(*msg#*, *buyer*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*) to LPs. The (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*) is a list of packaging tuples, each of which specifies that there are *quantity* of *load*s to be picked up from *origin* at *pickuptime* and to be delivered to *destination* at *arrivaltime*. For the first message from a buyer to an LP, or when the buyer seeks an updated quote, the *cost* will be set to null. Upon receiving the message, the LP consults its operations management system to compute the cost of the delivery job. It then sends the quote to the buyer with the message *L2B*(*msg#*, *LP*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*). The quote will last for *expirytime*. If the buyer is satisfied with the quote, it finalises the deal by sending the message *B2LAck(msg#)* back to the LP. If the buyer seeks to make a counter-offer to an LP, it will set *load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime* and *cost* to its desired values. It then sends a message *B2L*(*msg#*, *buyer*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*) to the LP. If the LP is happy with it, it will reply with the message *L2BAck(msg#)*. Both buyer and LP are now contracted.

3.2 Decision Mechanism

At the top level, buyers have to obey a set of criteria in order to make decisions about their external activities.

We modify multi-attribute decision making scheme proposed by Keeney and Raiffa ([7]). Instead of having a fixed weight attached to each attribute, we propose to have a dynamic weight table for each attribute. The dynamic weight table associates a set of weights with a number of environment conditions in one-to-one manner. The environment conditions are the facts, which affect the weight, or importance, of an attribute. For example, the weight for time and price are equal to 1 in normal situation. In the face of a shortage of a raw material, the weight of time is 10 times more than that of the price. The environment conditions can be achieved from internal or external sources. They can be in form of relation among attributes and facts. Therefore, the near optimal is acceptable to experienced agents who know how to make good profit in the long run.

Strategic weight

Condition	Attribute	Weight
X_1	a_1	w_{11}
X_2	a_1	w_{12}
X_1	a_2	w_{21}
X_2	a_2	w_{22}

$U = \sum a_q w_{qp}$ Where condition X_p holds

Buyer's ranking table

Sellers' offers	LPs' offers	$\sum a_q w_{qp}$
(msg_1, S_1)	(msg_7, L_7)	$\sum a_{qS1} w_{qp} + \sum a_{qLy} w_{qp}$
(msg_2, S_1)	(msg_8, L_8)	$\sum a_{qS1} w_{qp} + \sum a_{qL8} w_{qp}$
(msg_3, S_2)	(msg_9, L_8)	$\sum a_{qS2} w_{qp} + \sum a_{qL8} w_{qp}$

Figure 1: Strategic weight table, utility calculation and ranking for the best combination of seller and LP.

Each agent maintains the strategic weight table, which counsels the agent a value of the weight of an attribute under a certain condition. The weight of an attribute is in a range of integer numbers. The agent controls values of the lower and upper bounds. A condition change can be a stock shortage, a financial concern, a complaint from customer, or a combination of them. An agent learns from its experience that how the changing environment affects its business and how to react. The results of the reactions will be analysed by the agent and it will update the range of the weights and their values under such conditions. It specifies in the table about a certain condition X_p expressively. Each attribute a_q affected by the condition X_p will be assigned a specific weight w_{iq} to it. For example, when "there is a stock shortage of a raw material g" (good $g \in G$), the weight of the attribute "price" under this condition will be decreased while the weight of lead time will be increased. The agent updates its strategic weight table every time it senses a change in environment. This allows agent to act wisely in dynamic environment. Among the four attributes, trust is a special decision criterion in the sense that it depends on other attributes such as time and quality promised by other agents. Agents maintain trust differently. Each buyer has a trust table keeping record of other agents, including other buyers, sellers and LPs, and their trusted values. Each agent in the trust agent is associated with an integer value from range [-1,1]. The value is updated over times. When a buyer is dealing with a new agent, it may give a low value to trust, e.g. 0.3 or 0.4 to the new agent. If the agent performs well, e.g., keeps up with schedule, provide high quality of good, etc., the buyer increases trust value for that agent. On the other hand, if an agent fails to keep up with its promise, the buyer decreases the trust value. All the offers made by sellers and LPs are kept in buyers' knowledge base. Whenever it needs to compute the utility, it checks for the current environment condition X_p and take the corresponding weight w_{pq} of each attribute a_q. It creates a new tuple (*msg#*, *agent*) for each active offer and create a new ranking table composed of three fields: sellers' offer (*msg#*, *agent*), LPs' offers (*msg#*, *agent*) and $\sum a_q w_{qp}$. The first two fields are merely reference to the exact offers. The third field is the utility for each combination of sellers and LPs offers. The agent computes the utility for each combination by combining utility of the seller's offer and that of the LP's offer. The utility of an offer can be achieved by $\sum a_q w_{qp}$, where a_q is the attribute value retrieved from the corresponding offer, and w_{pq} is the weight of the attribute a_q under condition X_p.

3.3 Buyer to Buyer Negotiation

Direction	Message
Bl®Bm	*Bl2Bm(msg#, buyer, good, quality, (availabletime, origin, quantity), price, expirytime)*
Bl¬Bm	*Bm2Bl(msg#, buyer, good, quality, (availabletime, origin, quantity), price, expirytime)*
Bl®Bm	*Bl2BmDel(msg#, buyer, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)*
Bl¬Bm	*Bm2BlDel(msg#, buyer, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)*
Bl®Bm	*Bl2BmAck(msg#)*
Bl¬Bm	*Bm2BlAck(msg#)*
Bl®Bm	*Bl2BmDelAck(msg#)*
Bl¬Bm	*Bm2BlDelAck(msg#)*
Note: *Bl* is the leading buyer, *Bm* is a member buyer.	

Table 2: Buyers' negotiation messages on buying goods.

Buyers can form coalition to share discount on prices and delivery costs. A buyer can join any appropriate coalition. A buyer can join a coalition for discount on price and join another coalition for saving delivery cost. This allows agents to maximise benefits freely. When a buyer wants to form a coalition of buyers, it becomes a leading buyer, who tries to negotiate among buyers to form a buying coalition then negotiates on behalf of the coalition to agree on the prices with sellers. Firstly, the leading agent consults with its knowledge about other buyers who might be interested. Agents are considered being interested if *i*) they used to buy the same good at around this time previously, *ii*) they are located nearby, or *iii*) randomly selected. The leading agent sends the message *Bl*2*Bm*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*) with value of *origin*, *quantity* and *price* set to null to all interested agents. The interested agents consider the invitation and if they find that they are in need of the same *good* of the same *quality* at *availabletime*, they will reply by sending the message *Bm*2*Bl*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*) with the *quantity* set to an appropriate value back to the leading buyer. After the expiry time, the leading buyer assumes that all the truly interested buyers have responded. It then sums up all the *quantity*s and sends the message *B*2*S*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*) to sellers. The sellers will reply with the message *S*2*B*(*msg#*, *seller*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*).

Upon receiving quotes from sellers, the leading buyer select the one with maximal utility, computed the same way discussed above. It then computes the price for each interested agents. Then the leading agent sends the message *Bl*2*Bm*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*) to interested agents. If the all interested agents agree, they send the message *Bm*2*BlAck(msg#)* and the coalition is formed. If any of them is not satisfied with the share, they may negotiate on the price by sending message *Bm*2*Bl*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*). The leading buyer may try to satisfy the demanding buyers by giving away its share on the discount (Kraus et al., 2004) up to a limit. Whenever it goes beyond that, the leading buyer send the message *B*2*S*(*msg#*, *buyer*, *good*, *quality*, (*availabletime*, *origin*, *quantity*), *price*, *expirytime*) with the price increased by the excess to the seller. If the seller accepts it, the leading buyer sends the *Bl*2*BmAck(msg#)* to all members. The buying coalition is now formed. All members are obliged to pay the bills. If the seller does not accept, the negotiation goes on, virtually similar to that between a single buyer to a single seller. If the time runs out, then all of them suffer loosing opportunity.

Any buyer can try to manage transportation for other agents. Those agents may have been already interested in buying within the same buying coalition or may be located nearby. The process is similar to forming buying coalition. The leading buyer sends message *Bl*2*BmDel*(*msg#*, *buyer*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*) with *load*, *quantity* and *cost* set to null to all interested agents. The recipient agents consider if the *origin*, *destination*, *pickuptime*, and *arrivaltime* are suitable with their plans. The interested agents sent message *Bm*2*BlDel*(*msg#*, *buyer*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*) with (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*) and *cost* set to appropriate values back to the leading buyer. The leading buyer collects all replies and send message *B*2*L*(*msg#*, *buyer*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*) with (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*) set to all members' requests and cost set to null to LPs. The LPs reply with message *L2B(msg#, LP, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)* with *cost* set to appropriate value. The leading buyer computes for the best offer and compute for the cost for each interested agent. It then sends to interested agents the message *Bl*2*BmDel*(*msg#*, *buyer*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*) with *cost* set to their costs. If all the interested agents are happy with their costs, they reply with message *Bm*2*BlDelAck(msg#)*. The leading buyer sends the acknowledge message to the LP. The deal is done. If the interested agents are not satisfied with their costs, they negotiate with the leading agent by sending message *Bm*2*BlDel*(*msg#*, *buyer*, (*load*, *quantity*, *origin*, *destination*, *pickuptime*, *arrivaltime*), *cost*, *expirytime*) with their preferred *cost*s. The leading agent can try to negotiate with the unhappy buyers by giving away its share on the saved cost up to a threshold. If the demanding goes beyond the threshold, the leading buyer negotiates with the LP. The process repeats backward and forward until every agent is satisfied or the time expires.

3.4 Seller to Seller and LP to LP Negotiation

Sellers can form coalition when appropriate. Once a seller receives a quote it considers out of its resources or considers outsourcing is beneficial, it negotiates with other sellers. It firstly consults with its own plan for its capability and consults with its knowledge base for outsourcing to other sellers. It computes for other sellers' share on goods and prices. It sends message *Sl*2*Sm*(*msg#*, *seller*, *good*, *quality*, *availabletime*, *quantity*, *price*, *expirytime*) with price set to null to other sellers who can mutually cover the outsourcing. The recipients reply with the *price*s-quoted message *Sm*2*Sl*(*msg#*, *seller*, *good*, *quality*, *availabletime*, *quantity*, *price*, *expirytime*). If the leading seller is satisfied with all quotes, its finalises the deals with other agents by sending message *Sl*2*SmAck(msg#)*. If it is not satisfied with any of them, it can negotiate, similarly to how the leading buyer does it. Once all of the unsatisfied agents agree, they accept by sending message *Sm*2*SlAck(msg#)* back to the leader. Then the deal is done.

Direction	Message
Sl®Sm	*Sl2Sm(msg#, seller, good, quality, availabletime, quantity, price, expirytime)*
Sl¬Sm	*Sm2Sl(msg#, seller, good, quality, availabletime, quantity, price, expirytime)*
Ll®Lm	*Ll2Lm(msg#, LP, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)*
Ll¬Lm	*Lm2Ll(msg#, LP, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)*
Sl®Sm	*Sl2SmAck(msg#)*
Sl¬Sm	*Sm2SlAck(msg#)*
Ll®Lm	*Ll2LmAck(msg#)*
Ll¬Lm	*Lm2LlAck(msg#)*
Note: *Sl* is the leading seller, *Sm* is a member seller. *Ll* is leading LP, *Lm* is member LP.	

Table 3: Sellers' negotiation messages over selling goods and LPs' negotiation messages over delivery jobs.

When contacted from either a buyer or a seller, an LP may form coalition with others when it does not have enough resources or it finds that forming coalition is more profitable. The leading LP may divide whole distribution area into sections and decide to bilaterally negotiate with LPs, each of which is considered having enough potential to distribute the good on assigned section. The leading LP sends message *Ll2Lm(msg#, LP, (load, quantity, origin, destination, pickuptime, arrivaltime), cost, expirytime)* with *(load, quantity, origin, destination, pickuptime, arrivaltime)* set to calculated values to potential agents. Each member agent tries to accommodate the on-negotiating job with its own resources and its on-going plans. If the member is satisfied with the offer, it sends message *Lm2LlAck(msg#)* back to the leading LP—the deal is done. If not, it negotiates with the leading LP similar to buyers and sellers. Some LPs may not be satisfied with bilateral negotiation, they may force the leading LP to manage the multilateral negotiation for the sake of fairness and efficiency. LP negotiation is concerned with the cost of doing the job and the profit. The cost is involved with the route a LP run regularly and the route for the new job.

4. RELATED WORK

Coalition formation is an active research area in multi-agent systems. Previous work has done some part of our proposing framework as the followings. Li et al. ([8]) addressed the combinatorial auction in coalition formation. Buyers submit their requests to the mediator who will allocate the good from sellers to them. Goldman et al. ([2]) searched for a strategy where sellers selected the most profitable deal while buyers looked for the most satisfiable sellers. Kraus et al. ([6]) proposed a compromise strategy for distributing profits among sellers in order to form coalition quickly. They found that agents who are willing to give away their profits actually earn more. Breban et al. ([1]) addressed the coalition formation based on trust among agents. Trust is used as a mechanism to enforce agents to commit themselves to the jobs as parts of the coalition. Hyodo et al. ([3]) addressed the optimal coalitions of buyers and sellers who are located in distributed sites. They deploy genetic algorithm to search for the optimal share over the discount. Klusch et al. ([5]) addressed problems in static coalition formation and proposed an algorithm for mobile agents to form coalition under dynamic environment. Agents can form overlapping coalitions.

5. CONCLUSION AND FUTURE WORK

We propose a simple framework, which involves negotiation protocol and decision mechanism. The negotiation protocol allows thorough communication, i.e., buyers to buyers, buyers to sellers, sellers to sellers, buyer to LPs, and LPs to LPs. The decision mechanism is a modification of Keeney et al.'s ([7]), which allows agents to adapt its decision to suit with changing environment. Agents can select appropriate weights for the environment in order to make suitable decision. The traditional utility-based decision mechanism seems inadequate in complex settings in real world. New approaches, e.g., Markov decision processes, Bayesian networks, CPNets, are suitable for more complex and dynamic environment. Under uncertain environment, agents have to learn from the past and project it to forecast what will happen in the future. Efficient utilization of knowledge base should help agents perform better in dynamic coalition formation. Based on this framework, we want to develop agents who are able to evolve strategies in order to enhance cooperation. In stead of modeling supply chain competitive games, we want to explore that under changing environment, how agents can adapt their strategies to cooperate and be successful at the end.

REFERENCES

[1] Breban, S. and J. Vassileva, "A coalition formation mechanism based on inter-agent trust relationships", *proceedings of the 1st AAMAS*, pp306-307, ACM Press, 2002.

[2] Goldman, C. V., S. Kraus, and O. Shehory, "Agent strategies: for sellers to satisfy purchase-orders, for buyers to select sellers", *proceedings of the 10th European Workshop on Maamaw' 01*, Annecy, France, 2001.

[3] Hyodo, M., T. Matsuo, and T. Ito, "An optimal coalition formation among buyer agents based on a genetic algorithm", *proceedings of IEA/AIE 2003*, pages 759-767, 2003.

[4] Kahan, J. and A. Rapaport, *Theory of Coalition Formation*, Lawrence Erlbaum Associates, London, 1984

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

[6] Kraus, S., O. Shehory, and G. Taase, "The advantages of compromising in coalition formation with incomplete information", *proceedings of AAMAS 04*, 2004.

[7] Keeney, R. L. and H. Raiffa, Decisions with Multiple Objectives: Preferences and Value Tradeoffs, John Wiley & Son, New York, 1976

[8] Li, C. and K. Sycara, "Algorithm for combinatorial coalition formation and payoff division in an electronic marketplace", *proceedings of AAMAS 02*.

[9] von Nuemann, J. and O. Morgenstern, , 3rd edition, Princeton University Press, Princeton, 1953.