Agent-Based Negotiation and Decision Making for Dynamic Supply Chain Formation

Minhong Wang[1, *], Huaiqing Wang[2], Doug Vogel[2], Kuldeep Kumar[3], Dickson K.W. Chiu[4]
[1] Division of Information and Technology Studies, The University of Hong Kong
magwang@hku.hk
[2] Department of Information Systems, City University of Hong Kong
{iswang, isdoug}@cityu.edu.hk
[3] College of Business Administration, Florida International University
kumark@fiu.edu
[4] Dickson Computer Systems, 7 Victory Avenue, Kowloon, Hong Kong
dicksonchiu@ieee.org

Abstract

Modern businesses are facing the challenge of effectively coordinating their supply chains from upstream to downstream services. It is a complex problem to search, schedule, and coordinate a set of services from a large number of service resources under various constraints and uncertainties. Existing approaches to this problem have relied on complete information regarding service requirements and resources, without adequately addressing the dynamics and uncertainties of the environments. The real-world situations are complicated as a result of ambiguity in the requirements of the services, the uncertainty of solutions from service providers, and the interdependencies among the services to be composed. This paper investigates the complexity of supply chain formation and proposes an agent-mediated coordination approach. Each agent works as a broker for each service type, dedicated to selecting solutions for each service as well as interacting with other agents in refining the decision making to achieve compatibility among the solutions. The coordination among agents concerns decision making at strategic, tactical, and operational level. At the strategic level, agents communicate and negotiate for supply chain formation; at the tactical level, argumentation is used by agents to communicate and understand the preferences and constraints of each other; at the operational level, different

[*] Corresponding author. Tel.: (852) 2859-2474. Email address: magwang@hku.hk

strategies are used for selecting the preferences. Based on this approach, a prototype has been implemented with simulated experiments highlighting the effectiveness of the approach.

Key words: Software agent; Supply chain management; Coordination; Negotiation; Distributed decision making; Constraint Satisfaction; Quality of Service (QoS)

1. Introduction

The management of supply chains represents a critical competency in today's global market. A supply chain is a worldwide network of organizations and their associated activities that work together to produce value for the customer. Firms in a supply chain network are more dependent on access to the resources of other firms. With the emergence of computer- and web-based technologies, e-business supply chain has emerged a key enabler for creating extra value and competitive capabilities of organizations [Christiaanse et al. 2000]. The focus of supply chain management has been shifted from production efficiency to supply network strategies. Temporary supply chain networks that are formed and driven by demand have sprung up and operated for the lifespan of the market opportunity [Kumar, 2001]. New business models have created, such as demand chains, virtual enterprises and electronic marketplaces, for supporting supply chains in web-based environments [Labarthe et al., 2007; Mowshowitz, 2002; Kaplan et al., 2000]. However, supply chain networks are confronted with business dynamics from supply and demand, complex and dynamic relationships between partners, and much shorter response times to react to changes.

How can supply chain partners form temporary alignments to quickly respond to market requirements as well as effectively utilize their competencies? The ability to select suitable partners and effectively utilize their resources throughout the chain is a key to successful supply chain networks [Kumar et al., 2007]. To achieve this, it is crucial to rapidly identify suitable resources (partners or services) and effectively coordinate them through the formation process, where various constraints such as service capacity, quality, cost, timeliness, and interdependencies between services need to be taken into account [Forget et al., 2008; Wang et al., 2008]. The constraint may be limited to an individual service, called an intra-service constraint (e.g., the quantity of the components to be purchased in a procurement service), or related to more than one service, called inter-service constraints (e.g., a manufacturing service should be scheduled to start after the procurement service is completed). A partial solution to an individual service does not have a global view and would not satisfy both intra-service and inter-service constraints. A partial solution is usually unable to take into account the constraints embedded in interdependencies among the services, very often resulting in incoherent and contradictory hypotheses and actions. Existing studies on this problem have focused on facilitating bilateral exchange between customers and suppliers, and have relied on complete information about resources and tasks without adequately capturing the dynamics and uncertainties of the operating environments. In reality, complex economic activities often involve multiple and interrelated exchange relationships [Wang et al., 2006]. It is a complex problem to select, schedule, and coordinate a set of services from a large number of resources under various constraints and even uncertainties. The complexity is mainly due to the ambiguity in determining the requirements of component services (e.g., schedule and budget); the uncertainty of solutions to component services (e.g., availability, capacity, and cost); and interdependencies among component services. The uncertainties and constraints may result in dynamisms and difficulties in searching, selecting, and coordinating the services. Given this observation, the main problem is to find a way to achieve coordination and coherence among the decisions of individual services in a supply chain network.

Relevant work in this area can be found in workflow or business process management, where problems in managing activities under various resource and operational constraints are being investigated [Eder et al., 1999; Senkul et al., 2005]. Other related work includes job-shop scheduling in production and supply chains, where constraint logic programming and advanced planning and scheduling have been used to deal with scheduling problems [Fox et al., 2000; Baumgaertel et al., 2003; Monteiro et al., 2007; Moyaux et al., 2007]. In relation to this, recent studies have placed more emphasis on complexity and dynamics of supply chains and development of networks of distributed and cooperative entities to improve agility and global performance of supply chains [Lendermann et al., 2001; Caridi et al., 2004]. The finding of these studies reveal that agent-based distributed planning and coordination provides clear advantages over centralized approaches for supply chains; however it still represents a major challenge to coordinate the independent planning entities in building a coherent and efficient supply chain plan. Most work has concentrated on scheduling and run-time checking of supply chains or workflows, especially in the context where the requirements of each service are determined and all available resources are known in advance. Such work suffers from uncertainties and dynamisms of operating environments, where more robust and adaptive strategies should be considered.

The objectives of this research are to investigate the complexity of supply chain formation in terms of service selection, coordination, and composition in dynamic and uncertain environments, and to examine the mechanism of agent-based coordination in dynamic supply chain formation, i.e., how agents act independently, collaboratively, and self-adaptively in distributed decision making to achieve coherence among the services. The coordination among agents concerns decision making at strategic, tactical, and operational level. At the strategic level, agents communicate and negotiate for supply chain formation; at the tactical level, argumentation is used by agents to communicate and understand the preferences and constraints of each other; at the operational level, different strategies are used for selecting the preferences.

The rest of the paper is organized as follows. Section 2 presents the background knowledge and reviews on related work. Section 3 introduces an overview of our multi-agent framework for dynamic supply chain formation. In section 4, we detail our mechanisms for multi-agent negotiation and decision making under our framework. Section 5 evaluates our approach with simulation experiments before we discuss the conclusion of this paper with future work.

2. An Agent-Based Negotiation and Decision Making Approach for Dynamic Supply Chain Formation

In this study, we propose an agent-based negotiation and decision making approach for achieving coordination and coherence among the decisions of individual services in a supply chain network. The use of intelligent software agents along the enterprise supply chain has been investigated by a number of researchers [Fox et al., 2000; Swaminathan et al., 1998; Caridi et al. 2004; Nissen et al., 2006]. It offers a new perspective of autonomous activity, interactivity, reactivity, and proactivity in an attempt to extend beyond speeding the communications, calculations, and routine computation in business interactions. In this work, an agent-based dynamic negotiation and decision making framework is proposed for resources coordination in supply chain formation. Each agent works as a broker of each service, dedicated to decisions for selecting individual service solutions, as well as negotiating with other agents in refining the decision making to achieve compatibility. However, the incomplete knowledge of service resources and solutions interwoven with various constraints may prevent the agents from moving toward a mutually satisfying solution. To improve the efficiency of the coordination, argumentation-based negotiation is used to guide the agents in refining their decision making according to the argument information from other agents. Each agent is allowed to argue and negotiate, and thereby achieve compatibility among all the agents' decisions.

2.1. Agent-Based Approaches in Supply Chain Management

Automated decision making and coordination by software agents is a key enabling technology for electronic commerce. Software agents represent a software development paradigm where systems are viewed in terms of multiple autonomous and interacting problem-solving agents. A multi-agent system consists of a number of agents, which interact with one another in order to carry out tasks through cooperation, coordination and negotiation [Wooldridge et al., 1995]. A supply chain can be viewed as a network of autonomous business entities aimed at the procurement, manufacturing, and distribution of related products or services. Agent technology helps understand and model complex real-world problems and systems by concentrating on high-level abstractions of autonomous entities [Wang et al., 2004; 2005; Chiu et al., 2008]. The benefits of adopting agent technology in supply chains have been recognized in an increasingly wide variety of applications involving inter-enterprise collaboration, extending the boundaries of strategic partnership to wherever the network technologies can reach.

The pioneering work in [Fox et al., 2000; Swaminathan et al., 1998] identified fundamental entities for modeling supply chains. A number of recent studies have led to significant advances by placing more emphasis on complexity and dynamics of supply chains and developing networks of distributed and cooperative entities to improve agility and global performance of supply chains [Caridi et al., 2004]. Monteiro et al. (2007) addressed a hierarchical architecture to integrate individual planner agent, negotiator agent, and mediator agent with a decentralized control for achieving robustness and flexibility of the supply chain network. To model and simulate complex supply chains in a mass customization context, Labarthe et al. (2007) proposed a methodological framework based on an agent paradigm. Forget et al. (2008) explored a framework to design multi-agent behavior in a supply chain planning system, where agents were able to dynamically change their planning and coordination mechanism and, ultimately increase overall performance. Most work in this area has concerned with distributed planning and scheduling system that models the supply chain as a set of semi-autonomous and collaborative entities acting together to coordinate their decentralized plans [Lendermann et al., 2001; Baumgaertel et al., 2003]. In decentralized planning and coordination, communication and information sharing is an important strategy. Moyaux et al. (2007) proposed two principles to explain why and how companies should share information in order to reduce the fluctuations of order streams in supply chains. By modularizing a complex problem in terms of multiple autonomous components that can act and interact in flexible ways, agent technology is well suited for complex, dynamic, and distributed applications.

2.2. Argumentation in Multi-Agent Decision Making

One of the current factors fostering multi-agent systems development is the increasing popularity of Internet-based environments, which provides the basis for distributed problem solving where agents interact with each other to reach their individual or shared goals. In a multi-agent system, negotiation is the process by which a group of agents communicate in order to reach a mutually accepted agreement on some matter of common interest [Laasri et al., 1992]. However, the incomplete knowledge and the diverse conflicting influences present within a multi-agent society may prevent agents from incorporating all their social influences. In such situations, in order to function as a coherent society, agents require a mechanism to manage their social influences in a systematic manner. Various interaction and decision mechanisms have been proposed and studied, including game-theoretic analysis [Rosenschein et al., 1994; Kraus et al., 2001], heuristic-based methods [Faratin et al., 1998; Fatima et al., 2004], and argumentation-based approaches [Kraus et al., 1998; Parsons et al., 2003].

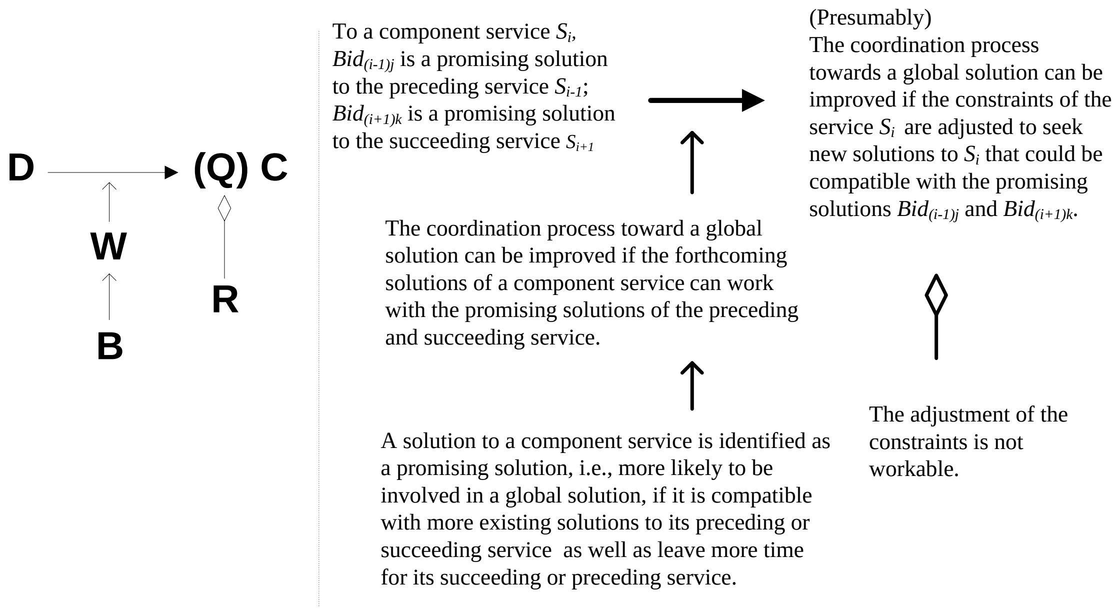
Among the various mechanisms, argumentation-based approaches attempt to overcome the limitations of other approaches by allowing agents to exchanging additional information to gain a wider understanding of each other’s capabilities, preferences, and constraints [Parsons et al., 2003; Rahwan et al., 2003; Bench-Capon et al., 2007]. Arguments of a proposal are a set of additional information that explains why the proposal should be accepted. Arguments enable agents to understand aspects of the preferences and constraints of other agents. In negotiation research, it has been proposed that performance of negotiation models can be improved through argumentation. Argumentation is now experiencing increased interests as a fundamental concept in multi-agent interaction. Parsons et al. (1996) provided a tighter integration of argumentation and the mental model of the negotiating agents. Kraus et al. (1998) presented a logical model of the mental states of the agents in argumentations based on a representation of their beliefs, desires, intentions, and goals; in this model, argumentation is regarded as an iterative process emerging from exchanges among agents to persuade each other and bring about a change in intentions for achieving cooperation and agreements. Based on the exiting studies in this area, Rahwan et al. (2003) proposed a conceptual framework to outline the core elements and features required by agents engaged in argumentation-based negotiation. In relation to supply chains, argumentation provides a rich and intuitive metaphor for interaction among distributed autonomous or semiautonomous entities [Rahwan et al., 2007]. Argumentation enables a richer form of negotiation of agents by arguing their beliefs and other mental attitudes during the decision making and coordination process. It has proven to be particularly suitable for dealing with reasoning under incomplete or contradictory information in a dynamically changing and networked distributed environment, such as supply chain networks.

2.3. Argumentation-Based Negotiation and Decision Making for Dynamic Supply Chain Formation

In e-supply chain formation, a set of services is selected, scheduled and coordinated under various uncertainties and constraints [Wang et al., 2008]. Each service broker agent is committed to selecting a solution to an individual service that should satisfy both intra- and inter-service constraints. However, these agents usually have problems to deal with inter-service constraints, which very often results in incoherent and contradictory hypotheses and proposals. Negotiation is required for achieving compatibility and coherence among the proposals. To solve the problem, the agents need to be able to exchange related information, and in particular, share the information so as to reach quicker decisions that are mutually satisfying. When there is a violation of constraints in the current solution of an agent, the agent may adjust its own solution requirements in order to find a feasible solution. However, if the received information places a hard constraint on the solution, the agent will override any local information that may cause conflicts. The process of acquiring information, resolving uncertainties, and revising preferences may improve the negotiation process by resolving potential conflicts among agents. In this way, the agents are able to resolve the conflicts of their local solutions and improve the coordination performance by considering such information in their decision making process.

In this study, each agent exchanges the candidate solutions of its own service with its neighboring (preceding or succeeding) service agents, as well as generates arguments with them. These arguments may provide useful feedback to the agents for guiding their decision making process and accordingly may improve the entire coordination process toward a global solution to the supply chain. In supply chain formation, a solution to a component service is more likely to be involved in a global solution if it connects with more existing solutions to its preceding or succeeding service, and if it leaves more time for its succeeding or preceding service. When such a solution has the potential to be involved in a global solution, we call it a “promising solution.” The information about a promising solution for a service is treated by the service agent as an argument, which is then passed to the neighboring service agents to guide their decision making in refining the constraints of their services to seek new solutions that could be compatible with the promising solution.

According to Toulmin’s argumentation model shown in Figure 1 (left side), we need to analyze six features of an argument: data (D), claim (C), warrant (W), backing (B), qualifier (Q), and rebuttal (R) [Toulmin, 2003]. The *data* are the facts (e.g., a car was assembled in Germany) that support the arguer’s *claim* (e.g., the car is an European Union product), while the *warrant* ensures the connection between data and claim (e.g., a product of Germany will generally be regarded as an European Union product), on the basis of some *backing* (e.g., relevant statutes and legal provisions); the qualifier specifies to what extent the warrant applies to the claim (e.g., presumably), and the *rebuttal* describes special conditions that undermine the warrant (e.g., most components of the product were manufactured out of Europe). Following this model, the framework of the argumentation mechanism proposed in this study is outlined in Figure 1 (right side). The details regarding why and how we use this mechanism are elaborated in the following sections.



To a component service S_i, $Bid_{(i-1)j}$ is a promising solution to the preceding service S_{i-1}; $Bid_{(i+1)k}$ is a promising solution to the succeeding service S_{i+1}

(Presumably) The coordination process towards a global solution can be improved if the constraints of the service S_i are adjusted to seek new solutions to S_i that could be compatible with the promising solutions $Bid_{(i-1)j}$ and $Bid_{(i+1)k}$.

The coordination process toward a global solution can be improved if the forthcoming solutions of a component service can work with the promising solutions of the preceding and succeeding service.

A solution to a component service is identified as a promising solution, i.e., more likely to be involved in a global solution, if it is compatible with more existing solutions to its preceding or succeeding service as well as leave more time for its succeeding or preceding service.

The adjustment of the constraints is not workable.

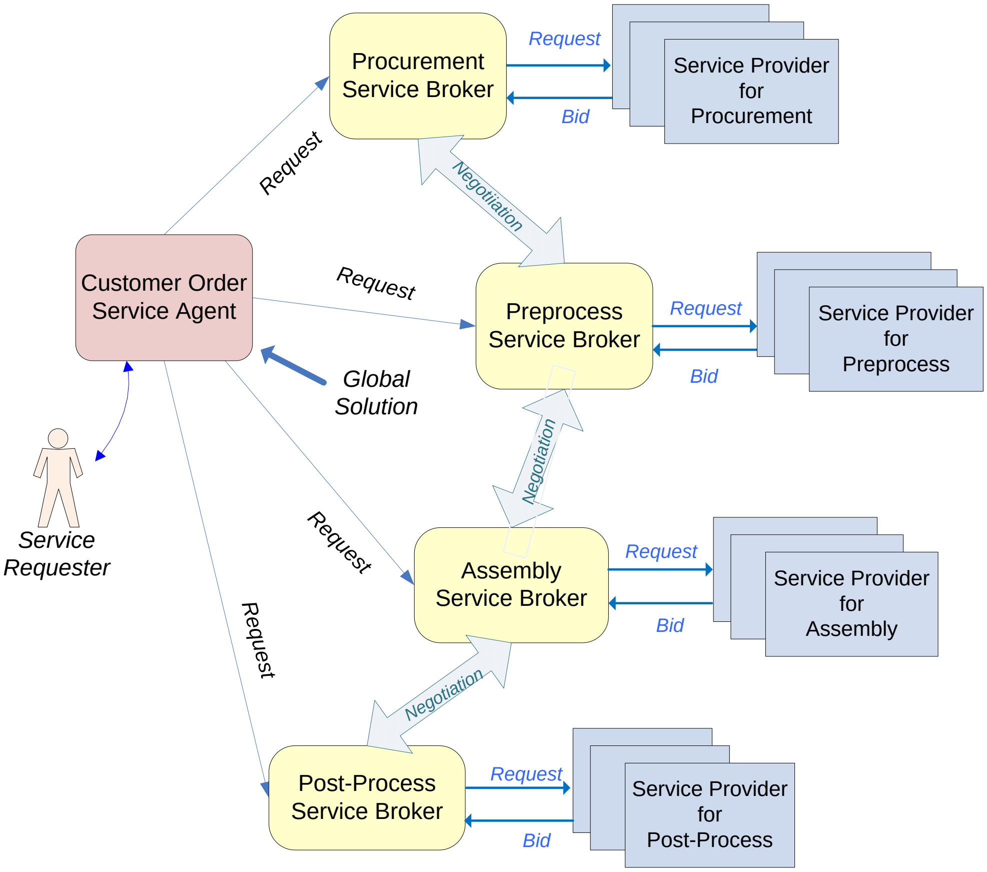
Figure 1. Argumentation model

The above model can be represented into formal symbolic systems or languages, such as predicate logic [Epstein, 2000]. With well-defined syntax, semantics and rules of inference, such kind of knowledge representation languages allow people to represent complex facts and make influence based on the facts. In this study, we focus on mechanism design of argumentation in agent-based negotiation, instead of the representation of the argumentation model using a formal symbolic language.

3. A Multi-Agent Framework for Dynamic Supply Chain Formation

In this work, supply chain formation is modeled as an agent-mediated decision making and coordination problem. As shown in Figure 2, a society of software agents, including a Service Requester, and a set of Service Brokers and Service Providers, is proposed. This corresponds to a supply chain, where a product service is fulfilled through a set of services including procuring components, preprocessing components, assembling components into products, post-processing products, and delivering components or products whenever the customer and service providers are distributed in different locations. For clarification, we use *component service* to indicate an individual service involved in the chain, and *composite service* to indicate an integrated service achieved through the supply chain.

With regard to a service request, the issues of time, cost, and location are considered important attributes of quality of service [Menasce, 2004]. Normally, a composite service should be completed on or before the due date and delivered to the location required by the customer; a feasible solution with the lowest cost will be accepted by the customer. Each component service should be scheduled to start after its *preceding service* is completed before its *succeeding service* starts. For clarity, we only consider the situations where there is only one preceding and succeeding service of a component service in this paper. Moreover, when the customer and component service providers are distributed in different locations, one or more delivery services are required to be embedded into the service chain.



Procurement Service Broker
Request
Bid
Service Provider for Procurement
Request
Negotiation
Customer Order Service Agent
Request
Preprocess Service Broker
Request
Bid
Service Provider for Preprocess
Global Solution
Negotiation
Service Requester
Request
Assembly Service Broker
Request
Bid
Service Provider for Assembly
Request
Negotiation
Post-Process Service Broker
Request
Bid
Service Provider for Post-Process

Figure 2. A multi-agent framework of e-supply chain formation

As discussed, we face both ambiguity in determining the requirements of component services and uncertainties of solutions to component services, which may further result in dynamics and difficulties in searching, selecting, and coordinating the solutions throughout the formation process. A major challenge of this work is that available resources of the component services (including service providers and their solutions) are unknown in advance. To find them, we need to send out service requests to service providers. However, we only have the constraints of the composite service and not the constraints of the component services. To solve the problem, we may first estimate the constraints of each component service and then refine this estimate based on real-time responses from service providers and real-time coordination among the services [Wang et al., 2008].

Upon receiving a request (e.g., 1000 products to XYZ Plaza before 25-02-2008) from a customer, the Service Requester will decompose the customer's request into a set of services with estimated constraints for each component service. The estimation is based on the customer's request as well as the history information of the component services. In this example, the estimation of time constraints is based on the average percentage of time spent on the services. The start time of the first service and the due time of the last service are taken as fixed, as required by the customer. Moreover, each service takes place in a location, and its output can be transported to another place (destination) for a succeeding service when necessary. The cost issue is considered in selecting cheaper partial solutions from providers; the total cost required by the customer will be considered at the final stage when combining the partial solutions toward a global solution.

Associated with the supply chain, a set of Service Brokers and Service Providers is deployed, each for a specific component service. To decrease the complexity of the coordination process caused by adding delivery services on demand, we treated delivery as a type of standard service that could be provided by a certain global delivery company (e.g., DHL), and could be bound with any component service when necessary. After receiving service requests from the Service Requester, Service Brokers will send them to Service Providers for collecting suitable solutions to each service. However, available solutions may be incompatible to form a global solution. A Service Broker needs to coordinate with its neighboring brokers to adjust the service requirements for seeking new bids that may reach a global solution.

4. Multi-Agent Negotiation and Decision Making for Dynamic Supply Chain Formation

In an argumentation-based multi-agent negotiation and decision making system, the agent should be equipped with the mechanisms to do the following: exchange information with other agents to update their understanding of the social situation; generate outgoing arguments to help other agents further understand the preferences and conflicts to improve the negotiation process; and at the same time respond to incoming arguments to update and adjust its own decision making process. In this study, the essential activities of Service Broker agents involved in the negotiation and decision making for supply chain formation are elaborated as follows.

4.1. Generating and Exchanging Partial Solutions

After receiving a service request from the Service Requester, each Service Broker may forward the request to corresponding Service Providers for collecting solutions, i.e., bids for the services. In order to reach a mutually accepted global solution, the agents need to exchange relevant information for negotiation and coordination. The information exchange should be minimized to consider communication cost, privacy policies, and system robustness. In this example, each Service Broker exchanges the information regarding the solutions to its own services with its neighboring Service Brokers.

We use S_i to denote a component service, and Rq_i to denote the requirement of the service. Rq_i is defined as follows:

$$Rq_i = [st_i, et_i, cost_i] \quad \textbf{(1)}$$

It consists of three parts: st_i denotes the start time scheduled for S_i; et_i denotes the due time for S_i; $cost_i$ denotes the expected cost of S_i.

A bid is regarded as a solution to a component service, which is defined as follows:

$$Bid_{ij} = [b\_id_{ij}, s\_t_{ij}, e\_t_{ij}, c_{ij}, loc_{ij}, des_{ij}] \quad \textbf{(2)}$$

Bid_{ij}, the *j*th bid sent to the Service Broker *i* for service *i* contains five parts: b_id_{ij} denotes the ID number of the bid; s_t_{ij} and e_t_{ij} denote the start time and end time, respectively, scheduled for the service; c_{ij} denotes the cost claimed by the Service Provider; loc_{ij} denotes the location of the service; and des_{ij} denotes the destination of the service. Each provider may generate a bid that satisfies the request at a lowest cost. In the case of no bid being generated due to the time constraints, the provider may relax the constraints.

For all bids proposed by Service Providers, the Service Broker will filter out dominated bids before posting candidate solutions. A newly received bid, $Bid_{i\beta}$ (bid β for service *i*), is identified as a dominated bid if it is not cheaper and more time-saving than an existing candidate solution Bid_{ij}, by satisfying the following condition:

$c_{i\beta} \geq c_{ij}$ AND $s\_t_{i\beta} \leq s\_t_{ij}$ AND $e\_t_{i\beta} \geq e\_t_{ij}$ AND $loc_{i\beta} = loc_{ij}$ AND $des_{i\beta} = des_{ij}$ **(3)**

Similarly, any existing solution Bid_{ij} will be dominated by a new bid $Bid_{i\beta}$ if the new bid is cheaper and more time-saving than the existing one by satisfying the following condition:

$c_{i\beta} < c_{ij}$ AND $s\_t_{i\beta} \geq s\_t_{ij}$ AND $e\_t_{i\beta} \leq e\_t_{ij}$ AND $loc_{i\beta} = loc_{ij}$ AND $des_{i\beta} = des_{ij}$ **(4)**

The information regarding candidate solutions to service *i* will be sent to the brokers of neighboring services, service *i+1* and service *i-1*, for information sharing. Meanwhile, service *i* will receive the information regarding candidate solutions of neighboring services from the neighboring (preceding and succeeding) Service Brokers.

4.2. Figuring Out Mutually Compatible Solutions

After receiving the bids from its neighboring Service Brokers, each broker will evaluate these bids and identify some of its own solutions that are compatible with the solutions of its neighbors. We denote service *u* and service *v* as the preceding and succeeding service of service *i*, and $Bid_{i\beta}$, $Bid_{u\alpha}$, and $Bid_{v\gamma}$ as bids of service *i*, service *u*, and service *v*, respectively. Service Broker *i* will connect $Bid_{i\beta}$ (a bid of its own service) with $Bid_{u\alpha}$ (a bid of its preceding service) if the two bids are compatible by satisfying the following condition:

$s\_t_{i\beta} > e\_t_{u\alpha}$ AND $loc_{i\beta} = des_{u\alpha}$ **(5)**

The Service Broker will also link $Bid_{i\beta}$ (a bid of its own service) with $Bid_{v\gamma}$ (a bid of its preceding service) if the two bids are compatible by satisfying the following condition:

$e\_t_{i\beta} < s\_t_{v\gamma}$ AND $des_{i\beta} = loc_{v\gamma}$ **(6)**

If the provider cannot make the service output reach the location of the succeeding service, a standard delivery service (e.g., DHL) could be bound to the service with delivery cost and delivery time taken into account.

An example is shown in Table 1. Each bid is posted with its start time, end time, and cost; the mutually compatible solutions are indicated by the connections among them. The end time and cost of the solutions have been adjusted when a delivery service is needed for transporting the output of a service to its succeeding service. The connection between the solutions will be removed if the conditions are violated (e.g., the cost is greatly increased, or the due time is delayed later than the start time of the succeeding service) after the adjustment by adding a delivery service. Based on the example in Table 1, we will provide further illustration of the negotiation mechanism in the following subsections and the experimental results in the next section.

Table 1. An example

Service		S_a	S_b	S_c	S_d
Current constraints (start time, end time, cost)		(1, 14, $1800)	(6, 18, $1400)	(14, 38, $2200)	(36, 40, $1300)
Candidate Bids (Bid_ID, start time, end time, cost)		(a2, 1, 14, $1650) (a7, 1, 7, $1800) (a11, 1, 8, $1790) (a12, 1, 12, $1680)	(b3, 12, 17, $1250) (b4, 7, 18, $1100) (b5, 12, 15, $1400) (b13, 6, 17, $1200)	(c1, 14, 37, $2100) (c3, 16, 38, $2150) (c6, 17, 39, $2200) (c9, 15, 37, $2120)	(d1, 37, 40, $1200) (d2, 38, 40, $1250) (d7, 36, 38, $1250)
Promising Solutions	PPS[†]	N.A.	(a7, 1, 7, $1800)	(b5, 12, 15, $1400)	(c3, 16, 38, $2150)
	SPS[‡]	(b5, 12, 15, $1400)	(c9, 15, 37, $2120)	(d2, 38, 40, $1250)	N.A.
Adjusted constraints (start time, end time, cost)		(1, 11, $2000)	(8, 14, $1800)	(16, 37, $2500)	(39, 40, $1500)

4.3. Generating Arguments for Global Solutions

In addition to generating outgoing proposals and evaluating incoming proposals, the agent needs to decide on responses or feedback to its counterpart for guiding the negotiation process. In this study, each Service Broker generates *promising solutions* as the feedback to its neighboring brokers, which may help them refine their decisions regarding how to more efficiently seek bids toward a global solution.

4.3.1. Arguments in the form of promising solutions

[†] PPS - Preceding Promising Solution, see more detail in section 4.3

[‡] ‡ SPS - Succeeding Promising Solution, see more detail in section 4.3

What is a *promising solution*? As mentioned earlier, those solutions with the potential to be involved in a global solution that satisfies both intra-service and inter-service constraints, we call promising solutions. In this example, a solution to a component service is more likely to be involved in a global solution if it connects with more existing solutions to its preceding or succeeding service, as well as if it leaves more time for its succeeding or preceding service. If a solution to a component service can work with more existing solutions to its preceding service as well as leave more time to its succeeding service, we call it a *Preceding Promising Solution*, or PPS; or if it can connect with more existing solutions to its succeeding service as well as leave more time for its preceding service, we call it a *Succeeding Promising Solution*, or SPS. As shown in Table 1, Bid_{b5} (b5, 12, 15, $1400) is identified as a PPS from the viewpoint of S_c. This is because Bid_{b5} is connected with more bids of its preceding service S_a and it leaves more time for its succeeding service S_c than the other bids of S_b. In this example, Bid_{b5} (b5, 12, 15, $1400) is also identified as an SPS from the viewpoint of S_a since Bid_{b5} is connected with more bids of its succeeding service S_c and it leaves more time for its preceding service S_a than the other bids of S_b.

In this approach, each Service Broker needs to evaluate all candidate solutions of its own service, from which to identify a PPS and pass the information to its succeeding Service Broker. At the same time, the Service Broker needs to identify a SPS and pass the information to its preceding Service Broker. In this way, each Service Broker may receive the information of PPS from its preceding Service Broker and the information of SPS from its succeeding Service Broker as the arguments. In Table 1, the Service Broker of S_c will receive the information of Bid_{b5} (b5, 12, 15, $1400) as its PPS and Bid_{d2} (d2, 38, 40, $1250) as its SPS. Accordingly, a new bid of S_c will more likely be involved in a global solution if it could be compatible with both Bid_{b5} and Bid_{d2}. This is because a new bid of the service will more probably be involved in a global solution if it is compatible with both the PPS from its preceding service and the SPS from its succeeding service.

A promising solution is identified on the basis of its promising value. The promising value of a solution is evaluated on the basis of its connectivity with its neighboring solutions and the free time it leaves for its neighboring solutions. In more detail, the preceding promising value (*Pre_prom*) of Bid_{ij} is evaluated by a weighted measurement as follows:

$$Pre\_prom\ (Bid_{ij}) = wp\_conn * Pre\_conn\ (Bid_{ij}) + wp\_tf\ \ * Pre\_tf\ (Bid_{ij}) \qquad \mathbf{(7)}$$

where $Pre\_conn\ (Bid_{ij})$ measures the connectivity of Bid_{ij} with its preceding solutions;

Pre_tf (Bid_{ij}) measures the free time Bid_{ij} leaves for its succeeding solutions; and *wp_conn* and *wp_tf* denote the weight of *Pre_conn* and *Pre_tf*, respectively. *Pre_conn* (Bid_{ij}) and *Pre_tf* (Bid_{ij}) are further detailed as follows:

$$Pre\_conn\ (Bid_{ij}) = (pre_{ij} - MINPRE_i) / (MAXPRE_i - MINPRE_i) \qquad \textbf{(8)}$$

where pre_{ij} denotes the number of the preceding bids that connect with Bid_{ij}; $MAXPRE_i$ denotes the maximum value of pre_{ij} for $\forall j$; and $MINPRE_i$ denotes the minimum value of pre_{ij} for $\forall j$.

$$Pre\_tf\ (Bid_{ij}) = (MAXET_i - e\_t_{ij}) / (MAXET_i - MINET_i) \qquad \textbf{(9)}$$

where e_t_{ij} denotes the end time of Bid_{ij}; $MAXET_i$ is the maximum value of e_t_{ij} for $\forall j$; and $MINET_i$ is the minimum value of e_t_{ij} for $\forall j$.

Similarly, the succeeding promising value (*Suc_prom*) of Bid_{ij} is measured by the following function:

$$Suc\_prom\ (Bid_{ij}) = ws\_conn * Suc\_conn\ (Bid_{ij}) + ws\_tf\ \ * Suc\_tf\ (Bid_{ij}) \qquad \textbf{(10)}$$

where *Suc_conn* (Bid_{ij}) denotes the function to measure the connectivity of Bid_{ij} with its succeeding solutions; *Suc_tf* (Bid_{ij}) measures the free time Bid_{ij} leaves for its preceding solutions; and *ws_conn* and *ws_tf* denote the weight of *Suc_conn* and *Suc_tf*, respectively. *Suc_conn* (Bid_{ij}) and *Suc_tf* (Bid_{ij}) are further detailed as follows:

$$Suc\_conn\ (Bid_{ij}) = (suc_{ij} - MINSUC_i) / (MAXSUC_i - MINSUC_i) \qquad \textbf{(11)}$$

where suc_{ij} denotes the number of succeeding solutions that connect with Bid_{ij}; $MAXSUC_i$ is the maximum value of suc_{ij} for $\forall j$; and $MINSUC_i$ is the minimum value of suc_{ij} for $\forall j$.

$$Suc\_tf\ (Bid_{ij}) = (MAXST_i - s\_t_{ij}\ ) / (MAXST_i - MINST_i\ ) \qquad \textbf{(12)}$$

where s_t_{ij} denotes the start time of Bid_{ij}; $MAXST_i$ denotes the maximum value of s_t_{ij} for $\forall j$; and $MINST_i$ denotes the minimum value of s_t_{ij} for $\forall j$.

Based on the above way to identify promising solutions, we illustrate another scenario presented in Table 1. From the viewpoint of S_b, Bid_{c9} (c9, 15, 37, $2120) is identified as a SPS (*Succeeding Promising Solution*) since it can connect with more existing solutions to its succeeding service S_d as well as leave more time for its preceding service S_b. On the other hand, Bid_{a7} (a7, 1, 7, $1800) is identified as a PPS (*Preceding Promising Solution*) from the viewpoint of S_b. This is because Bid_{a7} leaves more time to its succeeding service S_b; there is no preceding service for this bid. Accordingly, a new bid of S_b will more likely be involved in a global solution if it could be compatible with both Bid_{c9} and Bid_{a7}. This is because a new bid of the service S_b will more probably be involved in a global solution if it is compatible with both Bid_{a7} (the PPS for S_b) and Bid_{c9}, (the SPS for S_b).

The information of the PPS and SPS will be used for adjusting of the constraint of the service S_b.

Similarly, from the viewpoint of S_d, Bid_{c3} (c3, 16, 38, $2150) is identified as a PPS (*Preceding Promising Solution*). This is because Bid_{c3} can connect with more existing solutions to its preceding service S_b as well as leaves more time to its succeeding service S_d. On the other hand, there is no SPS (*Succeeding Promising Solution*) since S_d is at the end of service process. Accordingly, a new bid of S_d will more likely be involved in a global solution if it could be compatible with both Bid_{c3}. In this case, the due time requirement of the S_d should be taken into account for adjusting of the constraint of the service S_d, as shown in Table 1.

4.3.2. Selection of promising solutions as the arguments

In addition to argumentation generation, selection is another key step of argumentation. Given a number of candidate arguments that a service agent may utter to its counterpart, which is the most appropriate argument? In this study, promising bids are identified by each Service Broker as arguments to influence other Service Brokers in their decision making process. Normally, a promising solution, i.e., a bid of the highest promising value will be selected. However, a promising solution to an individual service may not necessarily be a good choice from the viewpoint of a global solution. The goal of supply chain integration is to design and operate the entire chain so that total chain-wide performance is maintained [Forget et al., 2008]. Function surfaces in such complex problems can be very rugged due to a number of dimensions. Finding a global solution to service integration is a challenge that needs to deal with multiple local optima. An agent pursuing its own objective may enter into conflict with other agents. The search for a global solution can easily get entrapped in local optima. A practical remedy to overcome the limitation is to diversify the search in the vicinity of local optima via random constructions; this attempts to avoid local optima by jumping out of them early in the computation [Rayward-Smith et al., 1996].

In this study, we used two strategies to select a promising solution and have tested each of their impacts on overall performance. Based on the promising value, a promising bid can be selected using either the *elitist* or *tournament* selection strategy. The elitist strategy selects the best bid, i.e., the bid of the highest promising value, of a component service. However, selecting a promising solution only from the viewpoint of an individual service may make the negotiation process entrapped in local minima. To avoid the problem, tournament selection is used as another strategy. Tournament selection is a useful and robust selection mechanism commonly used by genetic algorithms. It runs a "tournament" among a few individuals chosen at random from the population and selects the winner, i.e., the one with the best fitness. In this study, each Service Broker uses the tournament strategy to jump out of local optima via random construction. The broker randomly selects a set of bids, from which the strongest bid, i.e., the bid of the highest promising value, is selected as the promising solution. No recombination and mutation is operated on the bids after selection. In tournament selection, selection pressure can be easily adjusted by changing the tournament size. If the tournament size is larger, weak individuals have a smaller chance to be selected. Tournament selection is equivalent to elitist strategy when the tournament size is the population size (∞). In this way, the tournament size indicates how far the search is diversified from the local optima.

4.4. Responding to the Arguments

Based on the Promising Preceding Solution (PPS) and the Succeeding Promising Solution (SPS), the Service Broker may refine its service requirements to seek new bids that would be compatible with the promising solutions (PPS and SPS). After receiving $Bid_{u\alpha}$ (a solution of the preceding service S_u) as the PPS and $Bid_{v\gamma}$ (a solution of the succeeding service S_v) as the SPS, the Service Broker of S_i may adjust its service constraints or requirements Rq_i as follows.

$$Rq_i = [st_i, et_i, cost_i], \qquad \textbf{(13)}$$

where $st_i = e\_t_{u\alpha} + 1$; $et_i = s\_t_{v\gamma} - 1$ (1 indicates one time unit, for example, one workday). The expected cost of the service $cost_i$ may be increased by a certain percentage if the requested duration of the service is reduced after the adjustment, as shown in the example in Table 1.

If the adjustment generated on the basis of the promising solutions is not workable (e.g., when $e\_t_{u\alpha}+1 >= s\_t_{v\gamma} - 1$), the Service Broker may reject the argument and ask its neighboring Service Broker to recommend another choice for the promising solution. The Service Broker may keep its service constraints unchanged unless a workable argument is accepted. In this way, Service Brokers may achieve coordination and coherence among the decisions of component services through a series of negotiations and adjustments that are made individually, but in a coordinated fashion. Furthermore, Service Brokers may communicate with one another at regular intervals in figuring out a global solution. One or more feasible global solutions may be generated, and the one with the lowest cost would be reported to the Service Requester as a bid to the customer.

5. Evaluation

Based on the proposed approach, a prototype system has been implemented. Using this prototype, a number of experiments have been conducted to evaluate the feasibility and performance of the proposed approach. To the best of our knowledge, there is no other work investigating the problem addressed in this study. Most work has concentrated on scheduling and run-time checking of supply chains, especially in the context where the requirements of each service are predetermined and all available resources are known in advance [Caridi et al., 2004, Baumgaertel et al., 2003, Karageorgos et al., 2003]. Their approaches are not able to deal with the service composition problem, in which the complexity of the problem is considerably increased because we only know about the request of the composite service from the customer, and not the detailed requirements of the component services involved in the whole supply chain. The key point of the proposed approach lies in the way in which the Service Broker agents make decisions and coordinate in the composition process by identifying promising solutions to relevant services and using such information for guiding the refinement of service constraints.

The experiments simulate the agent-mediated supply chain formation environment described above. Based on the example illustrated in Figure 2 and Table 1, a set of software agents, including a Service Requester, and a set of Service Brokers and Service Providers, have been built. This corresponds to a supply chain, where a product service is fulfilled through a set of services including procuring components, preprocessing components, assembling components into products, and post-processing products. To test the approach, a number of customer requests are specified with different requirements for product quantity, total cost, and destination. Each request is multiplied by setting the due time of the customer request as very tight, tight, average, loose, or very loose.

The coordination process normally needs to go through a number of cycles of constraint refinement by each service broker for achieving a global solution. During each cycle, the constraints of each service will be adjusted by the service broker based on the promising solutions, as shown in Table 1. Based on the adjusted constraints, new bids will be collected from the service providers. For each request from the customer, the number of cycles of constraint refinement needed for achieving a global solution differs according to the customer request as well as the bids generated by the service providers. In the experiment, we will evaluate the success rate of the proposed approach, and at most 50 cycles of constraint refinement are allowed to go through for each customer request. Some requests can be satisfied with a global solution generated after 50 or less cycles, while some others remain unsatisfied at the end of the 50th cycle. The number of cycles used to achieve a global solution is recorded for measuring the speed of the coordination process. Each request is tested 500 times to calculate the success rate, average cost of achieved global solutions, and average cycles used to achieve a global solution. For each request, the environment is randomly initialized by varying the information of the procurement service providers in terms of price area, stock, deliverable time, and location, as well as by varying the information of preprocess, assembly, and post-process service providers in terms of price area, overall load, available time periods, and location. Four services (procurement, preprocess, assembly, and post-process) are to be composed in the supply chain, each associated with 24 providers. For each component service, a provider is supposed to submit a bid that satisfies the constraints at the lowest cost. However, a provider of the procurement service may generate more than one bid with different price and deliverable time; the earlier the deliverable time, the higher the price.

As discussed, the methods of evaluating and selecting promising solutions as arguments are regarded as a key success factor of the proposed approach, and therefore are the main element tested in the experiments. While evaluating a promising solution based on its connectivity with its neighboring service solutions and the time it leaves for them, it is found that either connectivity only (1/0) or free time only (0/1) cannot perform better than their mixed form, especially when the weight is about 0.35 for connectivity and 0.65 for time freedom. This weight structure also yields better results in other situations of due time; therefore we have chosen it for further experiments.

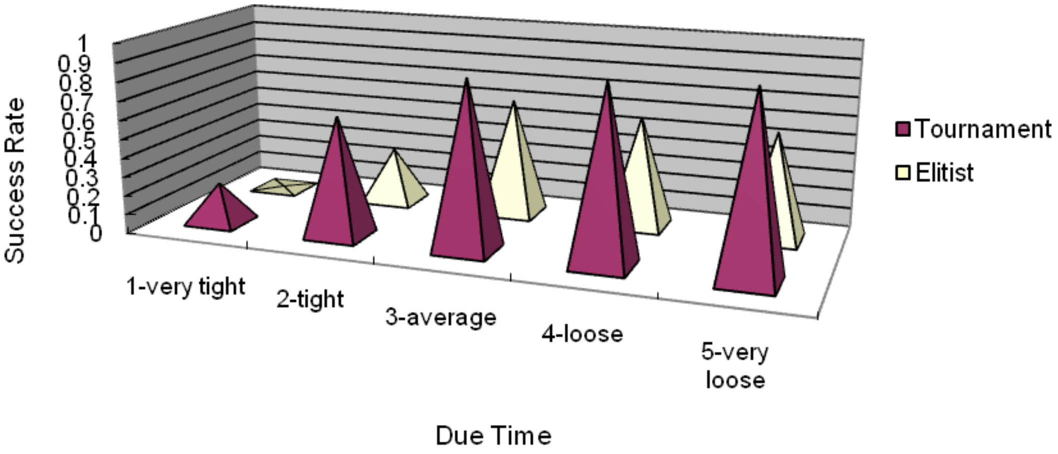


Figure 3. Comparison of the tournament and elitist selection strategies

The performance of the two strategies used for selecting a promising solution is summarized in Figure 3. The tournament selection strategy may exhibit different performances due to the tournament size. In the experiment of Figure 3, the tournament size is set as 4. For each component service, there are a number of candidate solutions collected from 24 service providers. From these candidate solutions, those compatible with other related services are identified. Among these identified compatible solutions, the elitist strategy selects the one of the highest promising value as the promising solution for the component service; the tournament selects 4 (the tournament size) of them randomly, from which the one of the highest promising value is select as the promising solution.

The result shows that the tournament selection strategy performs better than the elitist one with a higher success rate in most situations of due time. The difference is particularly significant when the composite service is required with a tight or very tight due date. The elitist strategy yields a poor success rate, mainly due to the premature convergence caused by the greedy nature of this strategy. The results have also shown that the elitist strategy has a slightly higher speed but a similar cost compared with the tournament strategy. This indicates that the elitist strategy may be faster at reaching a global solution if the searching is not entrapped in local optima.

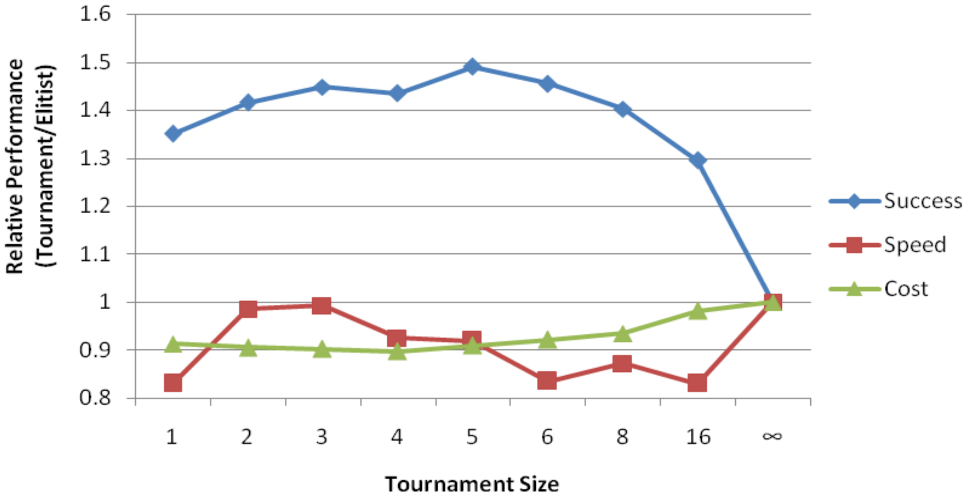


Figure 4. Performance of the tournament selection strategy

For the tournament selection strategy, different tournament sizes exhibit different performances. As shown in Figure 4, a number of tournament sizes are selected as 1, 2, 3, 4, 5, 6, 8, 16, and ∞ (the population size). Using different tournament sizes, we compare the tournament selection strategy with the elitist selection strategy by calculating their relative performances (tournament/elitist) with respect to success rate, cost, and speed.

The results of the experiments indicate that the tournament selection strategy yields an overall better performance with a higher success rate, and a higher speed when the tournament size is 3. When the tournament size is equal to the population size (∞), the tournament selection strategy is equal to the elitist strategy; when the tournament size is equal to 1, the tournament selection is equal to random selection.

6. Conclusion and Discussion

Supply chain formation is a complex task that requires the ability to search, schedule, and coordinate a set of services from a large number of service resources under various constraints and uncertainties. Decision making and coordination in supply chain formation is usually unstructured and partitioned into sub-problems, and software agents have become a key enabling technology for automated negotiation and decision making in this area. Existing approaches have been limited by concentrating on the situations where the requirements of each component service are determined and relevant resources and their status are known in advance. They have relied on complete information regarding service requirement and resources without adequately dealing with the dynamisms and uncertainties of the environments [Wang et al., 2008].

This study aims to support supply chain partners with a feasible solution for achieving coherence in a collaboration plan. We have proposed an approach to automate the supply chain formation in dynamic and uncertain environments, through agent-mediated negotiation and decision making. For clear presentation and ease of understanding, the paper has focused on the mechanism of automatic decision making and negotiation among software agents in service composition. Each agent works as a broker for each service type, dedicated to selecting solutions for each component service as well as interacting with other agents in refining the decision making to achieve compatibility. Finding a global solution to service composition requires that all agents coordinate and find the solutions that satisfy not only their own constraints, but also inter-agent constraints derived from interdependencies among services in a business process. Due to their irregular and ill-structured nature, systems for such collaborative problems may integrate the users into the problem-solving processes. This system can be viewed as an assistant, where managers of business entities interact with the system to confirm or modify the requests and the solutions proposed by the system. Human-based processes have historically had to deal with the complexity and need for flexibility, adaptability and spontaneity in the absence of comprehensive technological support. There are opportunities to learn from these dynamics in virtual teams in the quest for better technological support [Qureshi and Vogel, 2004; Rutkowski et al., 2002]. The effectiveness of the approach has been demonstrated via the experiments. Although presented in the context of supply chain management, this approach is appropriate to a wide variety of situations where a set of services is to be integrated as a result of a large number of resources to be searched, scheduled, and coordinated, especially in a real-time and adaptive fashion.

There are a number of implications of the study. The key point of the work is to understand the underlying mechanism of a complex system and model it by identifying autonomous and collaborative entities. It is extremely important to characterize agents, with their self-organization and interactive behaviors, using clear specifications. An agent’s activity and information flows must be designed in a manner that encapsulates complex processes and minimizes the need for coordination with other agents. Otherwise, the design specifications would be very complex and difficult to implement. It is also recommended that the whole system be designed as autonomous, i.e., with no central controller for directing and coordinating individuals. IT solutions for inter-enterprise integration need to provide a loose coupling between enterprises that allows for dynamic and peer-to-peer business relationships. Another implication of the study concerns decision making and coordination in a collaboration plan. The objective of a collaborative problem-solving system, such as the one we propose, is not to find an optimal solution, but to be able to formulate the alternatives, among which there may exist a satisfactory or feasible solution. Strategic management of decision making to reach feasible solutions in an effective and efficient way is also an important issue that we have begun to address. Meanwhile, we are interested in extending the techniques for mobile and ubiquitous service composition [Chiu et al., 2008].

Acknowledgement

This research is supported by UGC GRF Grant (716907 and 717708) from the Hong Kong SAR Government and Seeding Funding for Basic Research (200711159052) from The University of Hong Kong.

References

Baumgaertel, H., John, H., 2003. Combining agent-based supply net simulation and constraint technology for highly efficient simulation of supply networks using APS systems, *Proceedings of the 2003 Winter Simulation Conference*

Bench-Capon, T.J.M., Dunne, P.E., 2007. Argumentation in Artificial Intelligence, *Artificial Intelligence*, 171 (10–15), 619–641

Caridi, M., Cavalieri, S., 2004. Multi-agent systems in production planning and control: an overview. *Knowledge Information Systems*, 15 (2), 106-118

Chiu, D.K.W., Yueh, Y.T.F., Leung, H-f, Hung, P.C.K., 2008. Towards Ubiquitous Tourist Service Coordination and Process Integration: a Collaborative Travel Agent System with Semantic Web Services. *Information Systems Frontier*. DOI: 10.1007/s10796-008-9087-2

Christiaanse, E. and Kumar, K., 2000. ICT-enabled coordination of dynamic supply webs, *International Journal of Physical Distribution & Logistics Management*, 30 (3/4), 2000, 268-285

Eder, J., Panagos, E., Rabinovich, M., 1999. Time Constraints in Workflow Systems, *11th International Conference on Advanced Information Systems Engineering (CAiSE)*, LNCS 1626, 286-300

Epstein, J.L., 2001. *Predicate logic: the semantic foundations of logic***,** Belmont, CA: Wadsworth Thomson Learning

Faratin, P., Sierra, C., Jennings, N. R., 1998. Negotiation decision functions for autonomous agents, *International Journal of Robotics and Autonomous Systems* 24(3–4), 159–182

Fatima, S., Wooldridge, M., Jennings, N. R., 2004. An agenda based framework for multi-issues negotiation, *Artificial Intelligence Journal*, 152(1), 1–45.

Forget, P., D'Amours, S., Frayret, J.M., Gaudreault, J., 2008. Design of Multi-Behavior Agents for Supply Chain Planning: An Application to the Lumber Industry, In Kordic, V. (Ed), *Supply Chain, Theory and Applications*, I-Tech Education and Publishing, Vienna, Austria, 558-568

Fox, M.S., Barbuceanu, M., Teigen, R., 2000. Agent-Oriented Supply-Chain Management, *International Journal of Flexible Manufacturing Systems*, 12(2/3), 165-188

Kaplan, S., Sawhney, M., 2000. E-Hubs: The New B2B Market-places, *Harvard Business Review*, 78(3) 97-100

Karageorgos, A., Mehandjiev, N., Weichhart, G., Hämmerle, A., 2003. Agent-based optimisation of logistics and production planning, *Engineering Applications of Artificial Intelligence*, 16 (4), 335-348

Kraus, S., Sycara, K., Evenchik, A., 1998. Reaching agreements through argumentation: a logical model and implementation, *Artificial Intelligence*, 104 (1–2), 1–69

Kraus, S., 2001. *Strategic Negotiation in Multi-Agent Environments*, Cambridge, MA:

MIT Press

Kumar, K., 2001. Technology for supporting supply chain management: introduction, *Communications of the ACM*, 44(6), 58-61

Kumar, K. and Becerra-Fernandez, I., 2007. Interaction technology: Speech act based information technology support for building collaborative relationships and trust, *Decision Support Systems*, 43(2), 584-606

Laasri, B., Laasri, H., Lander, S. Lesser, V., 1992. A generic model for intelligent negotiating agents, *International Journal of Intelligent and Cooperative Information Systems*, 1, 291-317

Labarthe, O., Espinasse, B., Ferrarini, A., and Montreuil B., 2007. Toward a Methodological Framework for Agent-Based Modelling and Simulation of Supply Chains in a Mass Customization Context, *Simulation Modelling Practice and Theory*, 15 (2), 113-136

Lendermann, P., Gan, B.P., McGinnis, L.F., 2001. Distributed simulation with incorporated APS procedures for high-fidelity supply chain optimization, *Proceedings of the 2001 Winter Simulation Conference*

Menasce, D.A., 2004. Composing Web Services: A QoS View, *IEEE Internet Computing*, 8(6), 88-90

Monteiro, T., Roy, D., Anciaux, D., 2007. Multi-site coordination using a multi-agents system, *Computers in Industry*, 58 (4), 367-377

Moyaux T., Chaib-draa, B., D'Amours S., 2007. Information Sharing as a Coordination Mechanism for Reducing the Bullwhip Effect in a Supply Chain, *IEEE Transactions on Systems, Man, and Cybernetics*, 37 (3), 396-409

Mowshowitz, A., 2002. *Virtual organization: toward a theory of societal transformation stimulated by information technology*, Westport, Connecticut: Quorum Books

Nissen, M.E., Sengupta, K., 2006. Incorporating Software Agents into Supply

Chains: Experimental Investigation with a Procurement Task, *MIS Quarterly*, 30(1), 145-166

Parsons, S., Jennings, N.R., 1996. Negotiation through argumentation - a preliminary report, *Proceedings of the International Conference on Multi Agent Systems*, 267-274

Parsons, S., McBurney, P., 2003. Argumentation-Based Communication between Agents, In Marc-Philippe Huget (Ed.), *Communication in Multiagent Systems*, Lecture Notes in Computer Science, 2650, Springer, 164-178

Qureshi, S. and Vogel, D., 2004. Adaptiveness in Virtual Teams: Organisational Challenges and Research Directions, *Group Decision and Negotiation*, 10(1), 27-46

Rahwan, I., Ramchurn, S.D., Jennings, N.R., McBurney, P., Parsons, S., Sonenberg L., 2003. Argumentation-based negotiation, *The Knowledge Engineering Review*, 18 (4), 343-375

Rahwan, I., McBurney, P., 2007. Guest Editors' Introduction: Argumentation Technology, *IEEE Intelligent Systems*, 22(6), 21-23

Rayward-Smith, V.J., Osman, I.H., Reeves, C.R., Smith, G.D., 1996. *Modern heuristic search methods*, Chichester: UNICOM

Rosenschein, J.S., Zlotkin. G., 1994. *Rules of Encounter*, MIT Press, Cambridge MA

Rutkowski, A., Vogel, D., Genuchten, M., Bemelmans, T. and Favier, M., 2002. E-Collaboration: The Reality of Virtuality, *IEEE Transactions on Professional Communication*, 45(4), 219-230

Senkul, S. and Toroslu, I.H., 2005. An architecture for workflow scheduling under resource allocation constraints, *Information Systems*, 30(5), 399-422

Swaminathan, J.M., Smith, S.F., Sadeh, N.M., 1998. Modeling supply chain dynamics: A multiagent approach, *Decision Sciences Journal*, 29 (3), 607-632

Toulmin, S.E., 2003. *The Uses of Argument*, Cambridge University Press, Cambridge

Wang, M., Wang, H., Xu, D., Wan, K.K., Vogel, D., 2004. A Web-service Agent-based Decision Support System for Securities Exception Management, *Expert Systems with Applications*, 27 (3), 439-450

Wang, M., Wang, H., Xu, D., 2005. The Design of Intelligent Workflow Monitoring with Agent Technology, *Knowledge-based Systems*, 18 (6), 257-266

Wang, M., Wang, H., 2006. From Process Logic to Business Logic -- A Cognitive Approach to Business Process Management, *Information & Management*, 43(2), 179-193

Wang, M., Liu, J., Wang, H., Cheung, W., Xie, X., 2008. On-demand E-Supply Chain Integration: A Multi-Agent Constraint-Based Approach, *Expert Systems with Applications*, 34(4), 2683-2692

Wooldridge, M., Jennings, N.R., 1995. Intelligent Agents: Theory and Practice, *Knowledge Engineering Review*, 10(2), 115-152