Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 34 (2008) 2683–2692
Expert Systems with Applications
www.elsevier.com/locate/eswa



# On-demand e-supply chain integration: A multi-agent constraint-based approach

Minhong Wang [a,*], Jiming Liu [b], Huaiqing Wang [c], William K. Cheung [d], Xiaofeng Xie [d]

[a] *Division of Information and Technology Studies, The University of Hong Kong, Hong Kong*
[b] *School of Computer Science, University of Windsor, Canada*
[c] *Department of Information Systems, City University of Hong Kong, Hong Kong*
[d] *Department of Computer Science, Hong Kong Baptist University, Hong Kong*

**Abstract**

With e-business emerging as a key enabler to drive supply chains, the focus of supply chain management has been shifted from production efficiency to customer-driven and partnership synchronization approaches. This strategic shift depends on the match between the demands and offerings that deliver the services. To achieve this, we need to coordinate the flow of information among the services, and link their business processes under various constraints. Existing approaches to this problem have relied on complete information of services and resources, and have failed to adequately address the dynamics and uncertainties of the operating environments. The real-world situation is complicated as a result of undetermined requirements of services involved in the chain, unpredictable solutions contributed by service providers, and dynamic selection and aggregation of solutions to services. This paper examines an agent-mediated approach to on-demand e-business supply chain integration. Each agent works as a service broker, exploring individual service decisions as well as interacting with each other for achieving compatibility and coherence among the decisions of all services. Based on the framework, a prototype has been implemented with simulated experiments highlighting the effectiveness of the approach.


*Keywords:* Intelligent agents; Supply chain integration; Electronic business; Constraints management

## 1. Introduction

A supply chain is a worldwide network of organizations and their associated activities that work together to produce value for the customer. Supply chain management represents a critical competency in today's global market. The emerging global economy has increasingly put a premium on the ability of firms to quickly and accurately evaluate new market opportunities, new products, and other strategic business decisions in coordination with potential supply chain partners (Gosain, Malhotra, & El Sawy, 2005; Kumar, 2001; Nissen & Sengupta, 2006). By adopting e-business approaches, supply chains can, more rapidly and effectively, reap the benefits of reduced costs, increased flexibility and faster response times. Nowadays, supply chain networks that are temporarily integrated and driven by demands have emerged and operated for the lifespan of the market opportunity. However, the widespread use of information technology to create electronic linkages among partners may also result in unintended adverse effects on supply chain flexibility (Gosain et al., 2005). Supply chains are confronted with increased business dynamics from both demand and supply, complex and dynamic relationships between partners, and much shorter response times to changes. As a result, the focus of supply chain management has been shifted from production efficiency to customer-driven and partnership synchronization approaches in today's globalized competitive market. A key to this strategic shift is high-level collaboration between supply chain partners. It requires companies to be able to effectively and efficiently coordinate their activities across the supply

* Corresponding author. Tel.: +852 2859 2474; fax: +852 2858 5649.
*E-mail address:* magwang@hkucc.hku.hk (M. Wang).

0957-4174/$ - see front matter 
doi:10.1016/j.eswa.2007.05.041

chains that are dynamically set up in response to market requirements. Along with the strategic change, new business models have emerged, such as demand chain, virtual enterprise and electronic marketplace, for supporting supply chains in web-based environments (Blackwell & Blackwell, 2001; Kaplan & Sawhney, 2000; Mowshowitz, 1997).

In this study, we focus on the online formation of supply chains – how supply chain partners may be joined in temporary alignments to quickly respond to customer demand and effectively utilize resources. In a web-based environment, the ability to rapidly identify suitable resources (partners) and effectively coordinate them through the chain is crucial to the success of supply chain integration. Real-time decision and coordination among partners is crucial for building a global solution to service integration where *constraints*, such as time precedence, duration, capacity and location, need to be satisfied (Wang, Cheung, Liu, Xie, & Lou, 2006). A partial solution related to an individual service involved in the chain does not have a complete view of the global solution, very often resulting in incoherent and contradictory hypotheses and actions. For example, a manufacturing service may fail for lack of prerequisite resources if it is scheduled to start before the procurement service is completed. Existing approaches to this kind of problem have focused on facilitating bilateral exchange between customers and suppliers, and have relied on complete information of resources and tasks without adequately capturing the dynamics and uncertainties of the operating environments. Actually, complex economic activities often involve interrelated and multiple exchange relationships. It is a complex problem to search, schedule and coordinate a set of services from a large number of resources under various constraints. The complexity is mainly due to the uncertainty in determining the requests (start time, end time, cost, etc.) of individual services, as well as the unpredictability (availability, capacity, price, etc.) of solutions from service providers. Given this observation, the main problem is to find a way to achieve coordination and coherence among decisions of individual services in a real-time and adaptive fashion.

In this paper, we propose *an agent-mediated, constraint-based decision and coordination approach* to supply chain integration in a web-based environment. Recently, an emerging and technology-enabled innovation that involves the use of intelligent software agents along the enterprise supply chain has been investigated by researchers (Nissen & Sengupta, 2006; Sadeh, Arunachalam, Eriksson, Finne, & Janson, 2003). Automated decision and coordination by software agents is a key enabling technology for electronic commerce. It offers a new perspective of autonomous activity, interactivity, reactivity and proactivity in an attempt to extend beyond speeding up the communications, calculations, and routine computation in business interactions (Wang & Wang, 2006; Wang, Wang, & Xu, 2005). In this work, a multi-agent supported dynamic decision and coordination framework is introduced. Each agent works as a broker of each service, dedicated to decisions of individual service solutions, as well as interacting with each other for refining the decision, to achieve compatibility with other agents. Coordination among services is modeled as a distributed constraint satisfaction problem in which solutions and constraints are distributed into individual services to be solved by a group of agents.

The remainder of the paper is organized as follows. Section 2 discusses the related work of supply chain management with e-business, agent technology, and constraints management in supply chain integration. A framework of agent-mediated e-business supply chain integration is proposed in Section 3. The constraint-based decision and coordination mechanism of agents is elaborated in Section 4. Section 5 presents a prototype with experimental results. Finally, the conclusion and discussion are summarized in Section 6.

## 2. Background

### 2.1. Supply chain management and e-business

Conventional supply chain strategies have focused on individual firms as the competitive units, and have enforced optimization techniques to support the decision of individual functional operations, such as demand forecasting, inventory management, transportation scheduling, etc. In recent years, a combination of economics, technology and market places has compelled companies to examine and reinvent their supply chain strategies. To stay competitive, enlightened companies strive to achieve greater coordination and collaboration among supply chain partners in an approach called *supply chain integration* (Gosain et al., 2005; Lee & Whang, 2001). Information technology, particularly the Internet, play a key role in furthering the goals of supply chain integration. E-business has been a powerful and compelling enabler of supply chain integration across a wide range of industries (Kaplan & Sawhney, 2000). Businesses can use the Internet to gain global visibility across their extended network of trading partners and help them respond quickly to a range of variables, from customer demand to resource shortages. As a result, many of the core supply chain concepts and principles, such as information integration, workflow coordination, multi-party collaboration, on-demand supply chain, etc., have been put into practice in a much more effective way. In fact, there is an explosion of e-business applications in developing new paradigms of supply chain management. However, most of them have only scratched the surface. New models and approaches are continuously being developed for new initiatives and initiatives and technology solutions in this area.

### 2.2. Intelligent agents in supply chain management

Software agents represent a software development paradigm which is appropriate for distributed problem solving.

The term ''agent'' denotes an encapsulated software-based computer system that has autonomy, social ability, reactivity, and pro-activity. 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, 2002). 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 software systems (Liu, Jin, & Tsui, 2004; Liu, Jin, & Tsui, 2005). 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 (Sadeh et al., 2003). The MAS paradigm seems to be a natural choice for supply chain management, intrinsically dealing with coordination and coherence among multiple actors. In recent years, the application of agents for managing supply chains has been investigated in a number of studies (Fox, Barbuceanu, & Teigen, 2000; Lin & Lin, 2006; Sadeh et al., 2003). The benefits of adopting agent technology in supply chains are recognized in various aspects including information sharing, collaborative operation, enterprise integration, dynamic configuration, etc. Endowed with extended communication capabilities by the advances in network technologies, multi-agent systems are being used in an increasingly wide variety of applications involving inter-enterprise collaboration, extending the boundaries of strategic partnership to wherever the network technologies can reach (Wang, Wang, Xu, Wan, & Vogel, 2004).

### 2.3. Supply chain integration with constraint management

Supply chain integration refers to a philosophy for integrating all the activities among and between supply chain partners in the life of a product or a service from the earliest source of raw materials to the ultimate customer (Kumar, 2001). The success of supply chain integration is highly dependent on the match between the requirements and offerings that deliver the services. To achieve this, the separation between the requirements and the actors who satisfy these requirements has been recognized as an essence of supply chain integration for dynamic assignment of resources to requests (Mowshowitz, 1997). In supply chain integration, once a process or plan is determined based on the requirements, the real-time selection, coordination and aggregation of service partners are crucial for building a global solution where service providers are searched, scheduled and coordinated under various constraints (Wang, Cheung, Liu, & Luo, 2006). Computer- and web-based technologies have made it practical for supply chains to be established by selecting and integrating partners from numerous resources in a real-time fashion. With respect to the large resource repository in the web-based environment, there is a need to identify appropriate service operators, especially by addressing non-functional properties, such as cost, timeliness, security, and dependability (Cheung, Liu, Tsang, & Wong, 2004). Furthermore, there is growing attention paid to this field in terms of constraint-driven web service composition and QoS (Quality of Service) aware service composition, where web service compositions have been translated into a certain type of constraint satisfaction problem (CSP) (Ardagna & Pernici, 2006; Canfora, Penta, Esposito, & Villani, 2005; Menasce, 2004). Other relevant studies include job-shop scheduling in production and supply chains, where constraint logic programming has been used to deal with scheduling problems (Fox et al., 2000; Sadeh et al., 2003; Sauer & Appelrath, 2003). Similar issues have also been investigated in workflow or business process management, where constraint problem is investigated for managing activities under various resource and operational constraints. Time/temporal constraints were addressed as a critical component early in workflow systems (Eder, Panagos, & Rabinovich, 1999; Marjanovic & Orlowska, 1999; Zhuge, Cheung, & Pung, 2001); resource (such as people, machine, and software) constraints have been investigated lately in Li, Yang, and Chen (2004). However, most of the above work has concentrated on the situations where the requirements of services are determined and all available resources are known in advance. They require a completely specified problem as input and have not been able to adequately deal with uncertainties and dynamics in the environment.

## 3. Agent-mediated e-supply chain integration framework

In supply chain integration, we need to coordinate the flow of information among the services across the chain and link their business processes under various constraints. As discussed, we face the ambiguity in determining the requirements and solutions of individual services involved in a service chain, which may further result in the uncertainties and dynamics in searching, coordinating and integrating the solutions throughout the integration process. In the following sections we focus on handling such kind of over-constrained and complicated problem by using constraint-based agents that are able to adaptively and interactively make the decisions of services throughout a supply chain. For clarification, we use *composite service* to indicate an integrated service provided by the supply chain, and *component service* to indicate an individual service involved in the chain.

In this work, supply chain integration is mapped as an agent-mediated decision and coordination problem. As shown in Fig. 1, a society of software agents, including a Service Dispatcher Agent and a set of Service Broker Agents and Service Provider Agents, is proposed. This corresponds to the scenario depicted in Fig. 2, where a supply chain is fulfilled through a set of services, including procuring components, preprocessing components, assembling components into products, post-processing products, and delivering components or products, where the customer and service providers are distributed in different locations. With regard to service requirements, the issues of time,

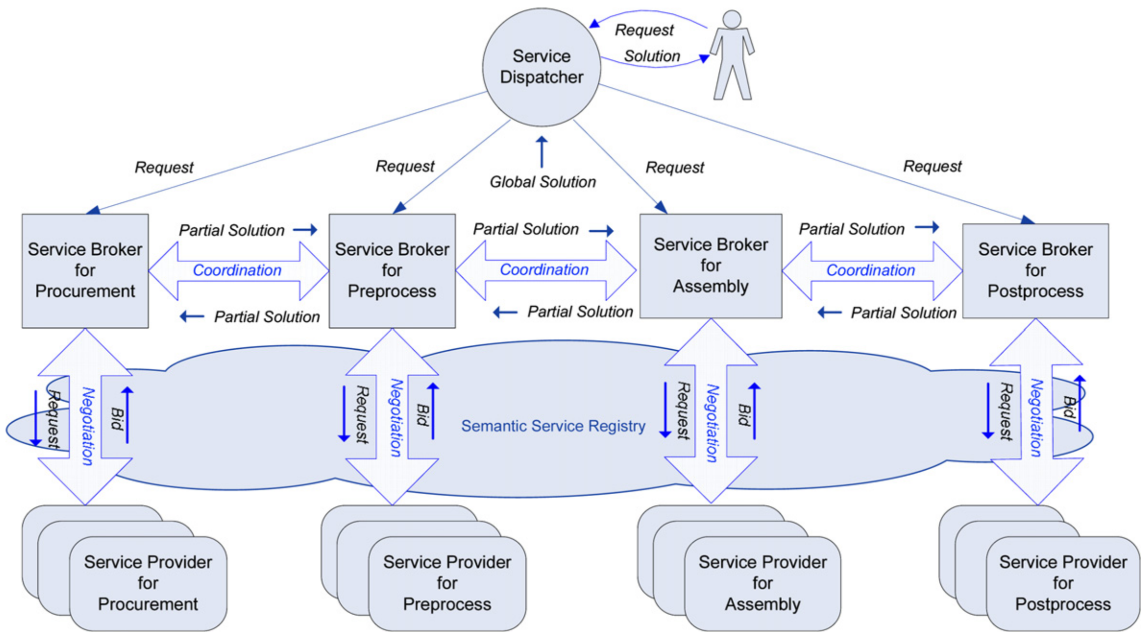


Fig. 1. Agent-mediated e-supply chain coordination framework.

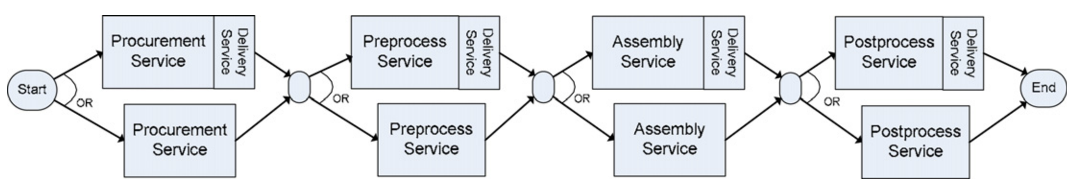


Fig. 2. A supply chain process.

cost, location, etc., are considered to be important attributes of quality of service (Menasce, 2004). Normally, a composite service is completed and delivered to the customer on or before the due date required by the customer and then a feasible solution with the lowest cost will be accepted. A component service is scheduled to start after its preceding service is completed, and to end before its succeeding service starts. Moreover, when the customer and component service providers are distributed in different locations, one or more delivery services are embedded into the process.

A major challenge of this work is that available resources of the component services (including service providers and their solutions) are *undetermined* in advance in most situations. To find them, service requests are sent out to service providers. However, only the requests of the composite service claimed by the customer are known, instead of the requirements of the composite services that constitute the chain. To solve the problem, the customer's request is transformed into a set of services and an estimate given for the requirements of these services, which can be further refined based on real-time responses from service providers and real-time coordination among services.

Upon receiving a request (e.g., 1000 products to XYZ Plaza before 25-02-2006) from a customer, the Service Dispatcher will decompose the customer's request into a set of services represented by a service plan/process associated with estimated requirements/constraints for each component service.[1] The estimation is based on the customer's request as well as the history information of the component services. The estimation of time constraints is based on the average percentage of time spent on the services and can be adjusted according to the history record. For example, a composite service $S$ is requested to start on July 1, and to finish by July 20. It is composed of three services, $S_1$, $S_2$ and $S_3$, taking 25%, 50% and 25% of time, respectively, on average. Accordingly, $S_2$ can be scheduled to start on July 6 and to end by July 15. Moreover, the start time of the first service and the end time of the last service is fixed, as required by the customer. Similarly, the locations and destinations of component services are initially specified by the Service Dispatcher and then tuned by Service Brokers, with the destination of the final service to be fixed.

Associated with the service chain, a set of Service Brokers is deployed, each for a specific component service. To decrease the complexity of the integration process

[1] This work focuses on the discussion of service coordination during service integration, therefore how to decompose the customer's request into a set of services is beyond the scope of this work.

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 Dispatcher, Service Brokers will send them to Service Providers for collecting suitable solutions to each service. However, available solutions may be incompatible with each other to form a global solution. A Service Broker needs to coordinate with its neighbouring brokers to refine the constraints for achieving new bids that could be involved in a global solution. The details of the coordination mechanism are illustrated in the following section. For the sake of simplicity, in this paper we ignore the situations where there is more than one preceding or succeeding service of a component service.

## 4. Agent-mediate decision and coordination mechanism

Decision and coordination in supply chain integration is usually complex and frequently partitioned into sub-problems. To deal with this, multi-agent theory proposes the kind of decomposition found in distributed decision making. During service integration, decision and coordination among services by agents are modeled as a distributed constraint satisfaction problem in which solutions and constraints are distributed into a set of services and to be solved by a group of agents. Finding a global solution to the composite service requires that all agents find the solutions that satisfy not only their own constraints but also inter-agent constraints (Liu, Jing, & Tang, 2002; Yokoo, 2001). The autonomous and interactive behavior of agents involved in decision and coordination for e-supply chain integration is elaborated as follows.

### 4.1. Collect solutions

After receiving the service request from Service Dispatcher, each Service Broker may forward the request to corresponding Service Providers for collecting solutions/bids of the service, where a bid is defined as follows:

$$Bid_{ij} = [b\_id_{ij}, s\_t_{ij}, e\_t_{ij}, c_{ij}, loc_{ij}, des_{ij}]$$

$Bid_{ij}$, the *j*th bid sent to Service Broker *i* for service *i* contains five parts: $b\_id_{ij}$ denotes the ID number of the bid, which is associated with the private details of a 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.

A bid is regarded as a partial solution corresponding to a component service. Each provider may generate a bid that satisfies the request with the lowest cost. In case of no bid generated due to the time constraints, the provider may relax the constraints as low as a bid generates. If the provider cannot make the service reach the destination, a standard delivery service could be bound to the service with delivery cost and delivery time taken into account.

### 4.2. Filter out dominated solutions

For all bids received from service providers, the Service Broker will filter out dominated bids before posting them as candidate solutions. A newly received bid, $Bid_{i\beta}$ (bid $\beta$ for service *i*), is identified as a dominated bid if it requires the same or higher cost as well as the same or more execution time compared with an existing solution $Bid_{ij}$, by satisfying the following condition:

$$c_{i\beta} \geqslant c_{ij} \text{ AND } s\_t_{i\beta} \leqslant s\_t_{ij} \text{ AND } e\_t_{i\beta} \geqslant e\_t_{ij} \text{ AND } loc_{i\beta} = loc_{ij} \text{ AND } des_{i\beta} = des_{ij}$$

Similarly, any existing solution $Bid_{ij}$ will be filtered out if it is dominated by a new bid $Bid_{i\beta}$ by satisfying the following condition:

$$c_{i\beta} < c_{ij} \text{ AND } s\_t_{i\beta} \geqslant s\_t_{ij} \text{ AND } e\_t_{i\beta} \leqslant e\_t_{ij} \text{ AND } loc_{i\beta} = loc_{ij} \text{ AND } des_{i\beta} = des_{ij}$$

After a bid is removed as a dominated bid, its connections with other bids are removed as well. By filtering out dominated solutions, the number of partial solutions is controlled in a reasonable scale.

### 4.3. Identify mutually compatible solutions

Each Service Broker will report its newly posted bids to its preceding and succeeding Service Broker, so that each broker may identify its solutions that are compatible with the solutions of its neighbours. We denote service *u* and service *v* as the preceding and succeeding service of service *i*, and $Bid_{i\beta}$, $Bid_{u\alpha}$, $Bid_{v\gamma}$ as a bid of service *i*, service *u*, service *v*, respectively. After posting $Bid_{i\beta}$, Service Broker *i* will connect it with $Bid_{u\alpha}$, an existing bid of its preceding service if the two bids are compatible by satisfying the following condition:

$$s\_t_{i\beta} > e\_t_{u\alpha} \text{ AND } loc_{i\beta} = des_{u\alpha}$$

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

$$e\_t_{i\beta} < s\_t_{v\gamma} \text{ AND } des_{i\beta} = loc_{v\gamma}$$

In the example shown in Fig. 3, where each bid is posted with its start time and end time, the mutually compatible solutions are connected to each other.

### 4.4. Figure out promising solutions towards a global solution

In service composition, each Service Broker may utilize its own information and limited information from its neighbours for coordinating and achieving coherence among the decisions. To achieve this, a Service Broker needs to identify a promising solution of its preceding

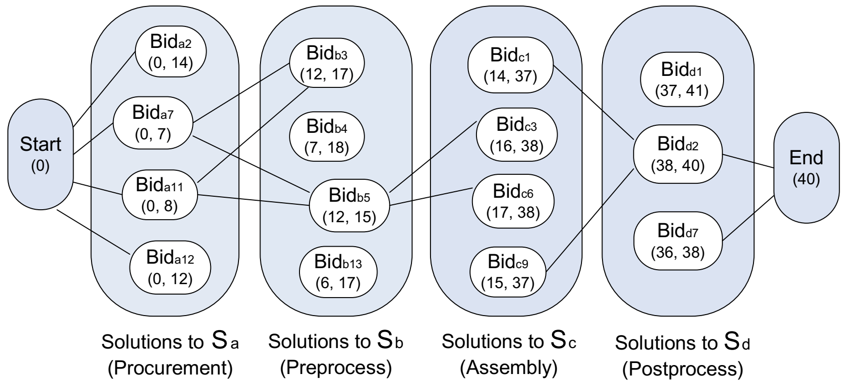


Fig. 3. Partial solution versus global solution.

service (*Preceding Promising Solution*, *P_PS*), as well as a promising solution of its succeeding service (*Succeeding Promising Solution*, *S_PS*). Based on this premise, the broker may refine the constraints of its own service to seek new bids that could work with the P_PS and S_PS. This is because a new bid of the service will probably be more involved in a global solution if it is compatible with the P_PS as well as the S_PS. Then, the problem is how to identify a promising solution (P_PS and S_PS) to a service.

What is a *promising solution*? A solution to a component service is more promising to be involved in a global solution if it can connect with more existing solutions to its preceding service as well as leave more free time to its succeeding service, or if it can connect with more existing solutions to its succeeding service as well as leave more free time to its preceding service. As shown in Fig. 3, $Bid_{b5}$ is identified as a P_PS from the viewpoint of $S_c$. This is because $Bid_{b5}$ is connected with more bids of its preceding service $S_a$ as well as leaving more time to its succeeding service $S_c$ than any other bid of $S_b$. On the other hand, $Bid_{b5}$ is also identified as a S_PS from the viewpoint of $S_a$ since $Bid_{b5}$ is connected with more bids of its succeeding service $S_c$ as well as leaving more free time for its preceding service $S_a$ than any other bid of $S_b$. In this way, each Service Broker may identify a P_PS among all the solutions of its preceding service, and a S_PS from all the solutions of its succeeding service. In Fig. 3, the Service Broker of $S_c$ will locate $Bid_{b5}$ as the P_PS, and $Bid_{d2}$ as the S_PS. Accordingly, a new bid of $S_c$ would probably be more involved in a global solution if it could connect with $Bid_{b5}$ as well as $Bid_{d2}$.

A promising solution is identified by its promising value. The promising value of a solution is measured based on its connectivity with its neighbour solutions and the free time it leaves for its neighbour solutions. In detail, the preceding promising value (*Pre_prom*) of $Bid_{ij}$ is measured by the following function:

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

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; *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:

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

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 $\forall j$; $MINPRE_i$ denotes the minimum value of $pre_{ij}$ for $\forall j$

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

where $e\_t_{ij}$ denotes the end time of $Bid_{ij}$; $MAXET_i$ is the maximum value of $e\_t_{ij}$ for $\forall j$; $MINET_i$ 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:

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

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; *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:

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

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

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

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

Based on the promising value, a promising bid can be selected by the Service Broker using different strategies, such as random selection strategy, elitist strategy and tournament selection strategy. The random selection strategy chooses a bid at random. The elitist strategy selects the best bid, i.e., the bid with the largest promising value. Tournament selection is one of many methods of selection in genetic algorithms which runs a "tournament" among a few individuals chosen at random from the population and selects the winner (the one with the best fitness) for crossover. Selection pressure can be easily adjusted by changing the tournament size. If the tournament size is higher, weak individuals have a smaller chance to be selected. Tournament selection is equivalent to random selection when the tournament size is 1, and equivalent to elitist strategy when the tournament size is the population size. As shown in Fig. 3, by using the elitist strategy, the Service Broker of $S_c$ may choose $Bid_{b5}$ as its P_PS, and $Bid_{d2}$ as its S_PS.

### 4.5. Refine constraints towards a global solution

After selecting $Bid_{u\alpha}$ (a solution of the preceding service $S_u$) as the P_PS and $Bid_{v\gamma}$ (a solution of the succeeding service $S_v$) as the S_PS, the Service Broker of $S_i$ may refine its service constraints $Rq_i$ as follows:

$$Rq_i = [st_i, et_i, loca_i, dest_i]$$

where
$st_i = e\_t_{u\alpha} + 1; et_i = s\_t_{v\gamma} - 1; loca_i = des_{u\alpha}; dest_i = loca_{v\gamma}$.

In this way, Service Brokers may achieve coordination and coherence among decisions of component services through a series of adjustments on constraints that are individually made but interact with each other. Furthermore, Service Brokers may communicate for figuring out a global solution at regular intervals. One or more feasible global solutions could be generated, and the one with the lowest cost reported as a bid to the customer.

## 5. Prototype and experiments

Based on the proposed approach, a prototype system has been developed. Developing of a software agent considers an internal model of the agent consisting of operational components, knowledge or information repositories, and correspondence with the external applications. At the outer layer, communication facilities allow an agent to exchange messages with the environment. The inner layer deals with models of action and behavior of an agent to execute different functions and collaborate with other agents. Information or knowledge is required by the agent to perform its internal and external activities. In this work, the focus of the implementation has been on developing the coordination modules of supply chain integration. The main activities of inter-agent coordination are illustrated through a sequence diagram in Fig. 4.

A number of experiments were conducted, and the results indicated the significant effectiveness of the approach. A service environment is randomly initialized by varying the distribution of price, stock, deliverable time, and location of the procurement service providers, as well as varying the distribution of price, overall load, available time periods, and location of the providers of production services such as preprocess, assembly and postprocess. Furthermore, each procurement service provider may have more than one service options associated with a different price, stock and deliverable time, e.g., the earlier deliverable time, the higher the price. For a procurement request, a supplier may generate a bid made of a pack of options

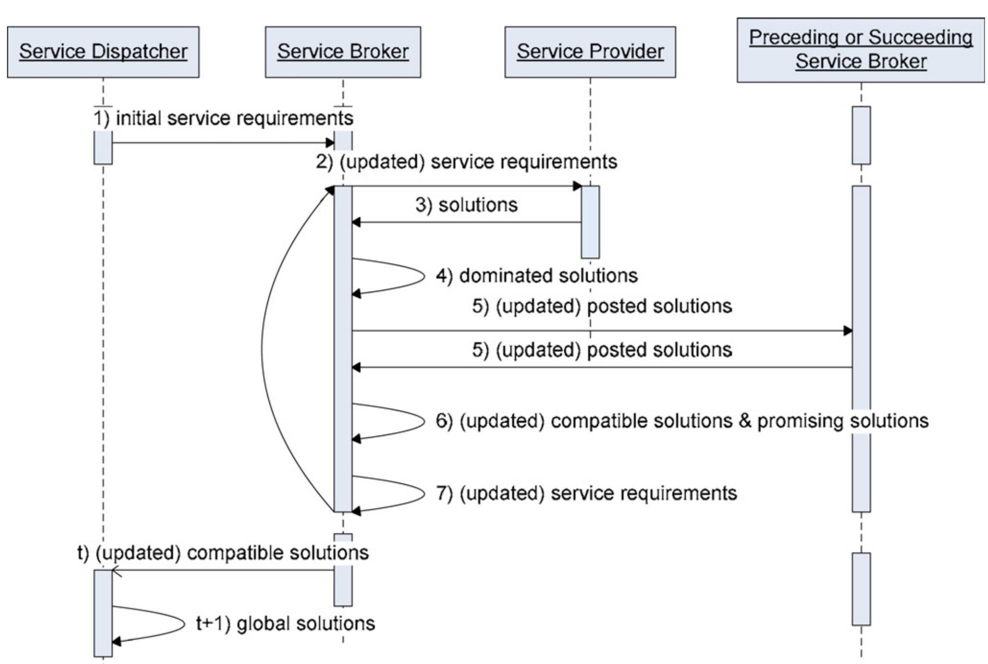


Fig. 4. Agent-mediated coordination activities.

that satisfy the time constraint at the lowest cost. The experiments simulate the agent-mediated decision and coordination in supply chain integration by associating 24 providers of each service type, and going through 50 cycles of constraint refinement for each problem. Each problem is tested in five sets of situations obtained by varying due time of the integrated service as very tight, tight, average, loose, and very loose. Each test is run 500 times to calculate the success rate, and average cost and average time used for achieving a global solution.

As discussed, the methods of evaluating and selecting a promising solution are regarded as the important success factor of the approach, and therefore are mainly tested in the experiments. While evaluating a promising solution based on its connectivity with its neighbour solutions and the free time it leaves for its neighbour solutions, 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 (see Fig. 5). Accordingly, the mixed form (0.35/0.65) is used in further experiments for comparing the three strategies of selecting a promising solution.

Fig. 6 summarizes the success rate of three strategies used for selecting a promising solution, i.e., random selection strategy, elitist strategy, and tournament selection strategy (when tournament size is 4). It is shown that the

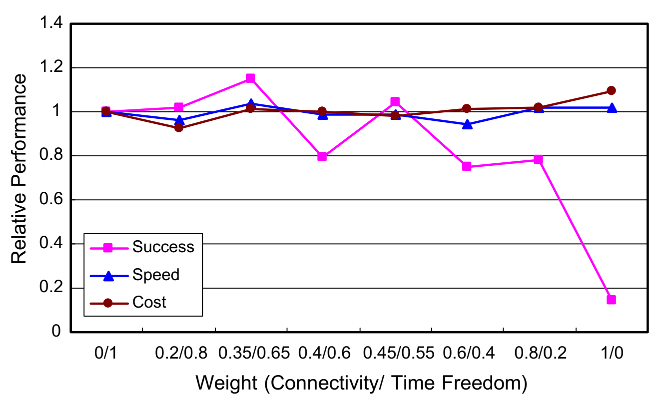


Fig. 5. Weight of connectivity and time freedom in the promising value.

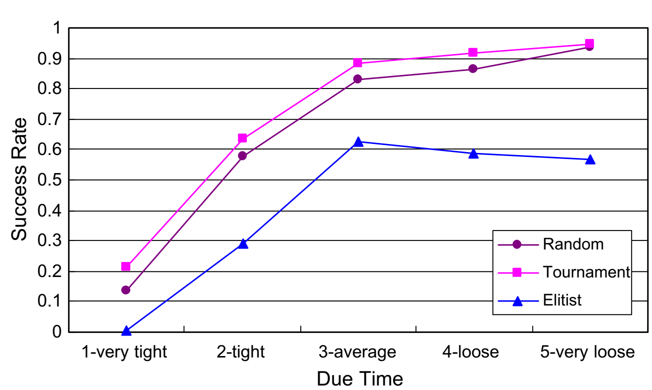


Fig. 6. Success rate of three selection strategies.

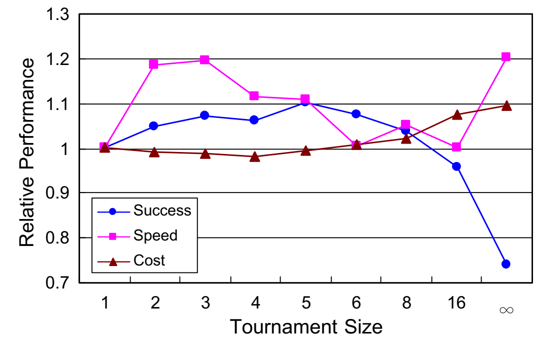


Fig. 7. Tournament size in tournament selection.

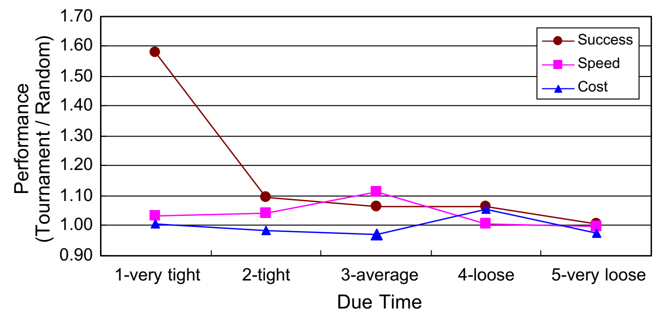


Fig. 8. Tournament selection versus random selection.

tournament selection strategy performs better than the other two by a higher success rate in most situations of due time. The experiments also show a faster speed and similar cost as a result of this strategy. The elitist strategy yields poor results, mainly due to the premature convergence caused by the greedy nature.

For the tournament selection strategy, different tournament sizes are tested. The tournament selection strategy yields an overall better performance when the tournament size is set as 4 or 5 in all situations of due time. As shown in Fig. 7, the success rate of the tournament selection strategy is about 10% higher than that of the random selection strategy when the tournament size is 5.

After setting the tournament size as 4, we compare the tournament selection strategy with the random selection strategy by calculating their relative performance. As shown in Fig. 8, the success rate of the tournament selection strategy is 58% higher than that of the random selection strategy in very tight due time situation. The speed of the tournament selection strategy is 11% faster than that of the random selection strategy in average due time situation. The results also indicate that the tournament selection strategy yields an overall better performance by more successfully generating a global solution with less time in most situations, however with less difference in cost.

## 6. Conclusion and discussion

E-business, the use of Internet-based computing and communications to execute business processes, has emerged as a key enabler to drive supply chain integration. By adopting e-business solutions, coordination and cooperation between competitors and partners, rather than optimizations of individual functions, has been a competitive success of modern supply chains. In such kind of environment, the ability to coordinate the flow of information among partners and to link business processes under various constraints is a key to supply chain success. Existing approaches to this problem have relied on complete information of all available resources and tasks, and cannot explicitly address the issues of uncertainties and dynamics as involved in a real-world operating environment. This paper has described a multi-agent framework that can be used to deal with dynamic and adaptive supply chain integration in a web-based environment. The contribution is to investigate the constraint-based decision and coordination among software agents in service allocation and integration through autonomous reasoning and constraint negotiation. The key idea lies in a distributed multi-agent system, in which autonomous agents have to make decisions of individual services to undertake within a supply chain. Coordination among agents is modeled as a distributed constraint satisfaction problem in which solutions and constraints are distributed into specific services. This system can self-organize itself, where each individual agent explores its own decisions, coordinates with other agents and gradually evolves towards a global solution state. 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.

The main objective of the approach is to support e-business partners to achieve feasibility and coherence in a collaboration plan. It tries to find a way to automate the integration process as much as possible, particularly to obtain coherence and coordination among decisions proposed by difference partners. Due to their irregular and ill-structured nature, systems for such kind of collaborative planning problems may integrate the users into the problem-solving processes (Carlsson & Turban, 2002). 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. For clear presentation and easy understanding, the paper has focused on the mechanism of automatic decision and coordination among software agents in e-supply chain integration. The interaction of the system with the users involved in the development of integration plans and its impact on the performance of the approach will be investigated in further studies.

## Acknowledgement

This work was supported by a RGC Central Allocation Group Research Grant (HKBU 2/03/C) from Hong Kong Government, and a Seed Funding for Basic Research (200611159216) from the University of Hong Kong. The author thank the editors and reviewers of Expert Systems with Applications, and Professor Kuldeep Kumar for their constructive comments on this paper.

## References

Ardagna, D., & Pernici, B. (2006). Global and local QoS guarantee in web service selection. In *Business process management workshops, LNCS 3812*, pp. 32–46.

Blackwell, R., & Blackwell, K. (2001). *The century of the consumer: Converting supply chains into demand chains, supply chain yearbook*. McGraw-Hill.

Canfora, G., Penta, M., Esposito, R., & Villani, M. L. (2005). QoS-aware replanning of composite web services. In *2005 IEEE international conference on web services (ICWS 2005)*.

Carlsson, C., & Turban, E. (2002). DSS: directions for the next decade. *Decision Support Systems, 33*(2), 105–110.

Cheung, W. K., Liu, J., Tsang, K., & Wong, R. (2004). Towards autonomous service composition in a grid environment. In *2004 IEEE international conference on web services (ICWS)*, pp. 550–557.

Eder, J., Panagos, E., & Rabinovich, M. (1999). Time constraints in Workflow systems. In *11th international conference on advanced information systems engineering (CAiSE), LNCS 1626*, pp. 286–300.

Fox, M. S., Barbuceanu, M., & Teigen, R. (2000). Agent-oriented supply-chain management. *International Journal of Flexible Manufacturing Systems, 12*(2/3), 165–188.

Gosain, S., Malhotra, A., & El Sawy, O. (2005). Coordination for flexibility in e-business supply chains. *Journal of Management Information Systems, 21*(3), 7–45.

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

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

Lee, H. L., & Whang, S. (2001). E-business and supply chain integration. Stanford Global Supply Chain Management Forum.

Li, H., Yang, Y., & Chen, T. Y. (2004). Resource constraints analysis of workflow specifications. *Journal of Systems and Software, 73*, 271–285.

Lin, F., & Lin, Y. (2006). Integrating multi-agent negotiation to resolve constraints in fulfilling supply chain orders. *Electronic Commerce Research and Applications, 5*(4), 313–323.

Liu, J., Jin, X. L., & Tsui, K. C. (2004). *Autonomy oriented computing (AOC): From problem solving to complex systems modeling*. Kluwer Academic Publishers/Springer.

Liu, J., Jin, X. L., & Tsui, K. C. (2005). Autonomy oriented computing (AOC): Formulating computational systems with autonomous components. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, 35*(6), 879–902.

Liu, J., Jing, H., & Tang, Y. Y. (2002). Multi-agent oriented constraint satisfaction. *Artificial Intelligence, 136*(1), 101–144.

Marjanovic, O., & Orlowska, M. E. (1999). On modeling and verification of temporal constraints in production workflows. *Knowledge Information Systems, 1*(2), 157–192.

Menasce, D. A. (2004). Composing web services: A QoS view. *IEEE Internet Computing, 8*(6), 88–90.

Mowshowitz, A. (1997). Virtual organization. *Communications of the ACM, 40*(9), 30–37.

Nissen, M. E., & Sengupta, K. (2006). Incorporating software agents into supply chains: Experimental investigation with a procurement task. *MIS Quarterly, 30*(1), 145–166.

Sadeh, N. M., Arunachalam, R., Eriksson, J., Finne, N., & Janson, S. (2003). TAC-03 – A supply-chain trading competition. *AI Magazine, 24*(1), 92–94.

Sauer, J., & Appelrath, H. (2003). Scheduling the supply chain by teams of agents. In *38th Hawaii international conference on system sciences (HICSS)*.

Wang, M., Cheung, W. K., Liu, J., & Luo, Z. (2006). Agent-based web service composition for supply chain management. In *IEEE joint conference on e-commerce technology (CEC' 06) and enterprise computing, e-commerce and e-services (EEE' 06)*, pp. 328–332.

Wang, M., Cheung, W. K., Liu, J., Xie, X., & Lou, Z. (2006). E-service/process composition through multi-agent constraint management. In *Fourth international conference on business process management (BPM), LNCS 4102*, pp. 274–289.

Wang, M., & Wang, H. (2006). From process logic to business logic – A cognitive approach to business process management. *Information and Management, 43*(2), 179–193.

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., 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.

Wooldridge, M. (2002). *An introduction to multiagent systems*. England: J. Wiley.

Yokoo, M. (2001). *Distributed constraint satisfaction: Foundation of cooperation in multi-agent systems*. Berlin, New York: Springer.

Zhuge, H., Cheung, T., & Pung, H. (2001). A timed workflow process model. *Journal of Systems and Software, 55*(3), 231–243.