Scientia Iranica, Vol. 15, No. 5, pp 623–636

SCIENTIA
IRANICA

A Fuzzy Intelligent Information Agent Architecture for Supply Chains

M.H. Fazel Zarandi[1,*], I.B. Turksen[2], S.M. Moattar Hoseini[1], S. Bastani[3] and A. Mohebi[1]

Through the emergence of information and communication technologies and customer-oriented approaches in business and industry, for achieving competitive advantages and in order to remain at the top in every business, more flexible and responsive supply chain systems are required. The next generation of supply chain systems must be agile, adaptive, cooperative, integrated and flexible. Agent-based supply chain management is an approach that addresses the next generation of supply chain system features. This paper focuses on the role of an information agent in agent-based supply chain management within an uncertain environment. For this purpose, a proper modular architecture for the information agent, based on fuzzy theory, is proposed. Here, the knowledge-based module in the architecture is fuzzy rules. The system is used for updating forecasted values and implementing customer commitment in a proper manner. Finally, the proposed architecture is tested and verified and the results of the developed approach are discussed.

INTRODUCTION

A supply chain encompasses the process from provision of the initial material to the ultimate consumption of the finished product, linking across supplier-user companies. It involves the functions within and outside a company that enable the value chain to make products and provide services to the customers [1]. Supply Chain Management (SCM) is a strategic approach that contains the following processes [2]:

- Customer relationship management,
- Customer service management,
- Demand management,
- Order fulfillment,
- Manufacturing flow management,
- Procurement,
- Product commercialization,
- Returns management.

Graves and Willems [3,4] developed an optimization algorithm to find the best inventory levels of all sites on the SC. They also extended their model to solve the supply chain configuration problems for new products. Cebi and Bayraktar [5] proposed an integrated Lexicographic Goal Programming (LGP) and AHP model, including both quantitative and qualitative conflicting factors for supply chains. Wang et al. [6] presented a weighted multiple criteria model for SC. They stated that, in real world problems, the weight of different criteria may vary, based on purchasing strategies. Stadtler [7] presents the main difficulties of SCM and tries to present some new models to resolve them. Baganha and Cohen [8], Graves [9], Chen et al. [10] and Li et al. [11] study the demand updating and information sharing issues.

Li et al. [11] use the term “information transformation” to describe the phenomenon where, for each considered stage, outgoing orders to the higher stage of a supply chain have different variance from the incoming orders that each stage receives.

1. *Department of Industrial Engineering, Amirkabir University of Technology, P.O. Box 15875-4413, Tehran, Iran.*
2. *Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON, Canada.*
3. *Department of Social Sciences and Economics, Azzahra University, Tehran, Iran.*
*. *To whom correspondence should be addressed: E-mail: zarandi@aku.ac.ir*

Through the emergence of new tools in information and communication technologies, globalization and in the shifting from mass production to mass customization, new requirements for achieving competitive advantages in supply chain management have been defined. These changes have led to the next generation of supply chain management systems. Such systems must have, at least, some essential characteristics, such as agility, responsiveness, adaptability, integration and cooperation [12,13]. The most effective areas that have drastically changed SCM are distributed artificial intelligence and agent-based systems.

In the literature, there are some research manuscripts that show Distributed Artificial Intelligence (DAI), especially agents and Multi-Agent Systems (MAS) for SC [14,15]. The multi-agent systems paradigm is a valid approach to modeling supply chain networks and for implementing supply chain management applications. Multi-agent computational environments are well-suited for analyzing coordination problems, involving multiple agents with distributed knowledge. Thus, a MAS model seems to be a natural choice for the next generation of SCM, which is intrinsically dealing with coordination and coherence among multiple factors [15,16]. The inherent autonomy of software agents enables the different business units of a supply chain network to retain their autonomy of information and control, and allows them to automate part of their interactions in the management of a common business process [17].

As uncertainty in the environment of a supply chain is usually unavoidable, an appropriate system is needed to handle it. Fuzzy system modeling shows its capability of addressing uncertainty in a supply chain. It can be used in an agent-based supply chain management system by the development of fuzzy agents and a fuzzy knowledge-base. Fuzzy agents use fuzzy knowledge-bases, fuzzy inference and fuzzy negotiation approaches to handle problems in the environment and take into consideration uncertainty. Using fuzzy concepts lead to more flexible, responsive and robust environment in the supply chain, which can handle changes more easily and cope with them more naturally.

Erol and Ferrel [18] discussed applications of the fuzzy set theory in finding the supplier with the best overall rating among suppliers. Fazel Zarandi and Saghiri [19] presented a fuzzy expert system model for SC complex problems. They compared the results of their proposed expert system model with fuzzy linear programming and showed its superiority. Zarandi et al. [20] presented a fuzzy multiple objective supplier selection model in multiple products and in the supplier environment. In their model, all goals, constraints, variables and coefficients are fuzzy. They showed that, by the application of fuzzy methodology, the multi-objective problem is converted to a single one.

MULTI AGENT SYSTEMS AND AGENT-BASED SUPPLY CHAIN MANAGEMENT

Software agents are just independently executed programs, which are capable of acting autonomously in the presence of expected and unexpected events [21]. To be described as intelligent, software agents should also process the ability of acting autonomously; that is, without human input at run-time and with flexibility; that is, being able to balance their reactive behavior in response to changes in their environment with proactive or goal-directed behavior [22]. These issues have also been discussed by other authors, which were classified by Liu et al. [23].

As stated by Fox et al. [21], in the context of multiple autonomously acting software agents, the agents additionally require the ability to communicate with other agents, that is, to be social. The ability of an agent to be social and to interact with other agents means that many systems can be viewed as Multi-Agent Systems (MAS). The hypothesis or goal of multi-agent systems is to create a system that interconnects separately developed agents, thus, enabling the ensemble to function beyond the capabilities of any singular agent in the system [21].

In multi-agent systems, some issues, such as agent communication, agent coordination and inference must be considered [24]. For agents to communicate with each other, an Agent Communication Language (ACL) is needed. Multi-agent systems have been applied to supply chain management and have introduced a new approach called agent-based supply chain management. In agent-based supply chain management, the supply chain is considered as being managed by a set of intelligent software agents, each responsible for one or more activities in the supply chain and each interacting with other agents in the planning and execution of their responsibilities.

For applying agents to supply chain management, first the following issues must be considered [12,13,17]:

- The distribution of activities and functions between software agents,
- Agent communication issues, including:
  - Interoperability,
  - Coordination,
  - Multi-agent scheduling and planning,
  - Cultural assumptions.
- Responsiveness,
- Knowledge accessibility in a module.

During the past decade, agent-based supply chain management has been the main concern of many researchers. Saycara [25] has done related projects and research in this area. Lambert et al. [12] introduce virtual supply chain management and a virtual situation room, in which agents are the main elements for achieving a coordinated and cooperative supply chain. Jiao et al. [26] propose the use of multi-agent system concepts in global supply chain networks. Xue et al. [27] suggest a framework for supply chain coordination in a construction networks. Wang and Sang [28] present a multi-agent framework for the logistics in a supply chain network. Fox and Barbuceanu [29] discuss a model for agent negotiation and conversation in agent-based supply chain management. Dasgupta et al. [30] focus on negotiations between suppliers at different stages in supply chain management. Chauhan [31] and Lau et al. [32] propose a methodology for multi-agent systems development in the supply chain. Chauhan [31] used Java technology and an object-oriented approach to achieve the goal. Lau et al. [32] introduce a methodology for a flexible workflow system in the supply chain, in order to obtain more flexibility in the ever-changing environment of the supply chain.

Some researchers present some architecture for agent-based supply chain management. Ulieru et al. [33] introduce a common architecture for collaborative Internet-based systems, in which some services are delivered via the Internet. The architecture was for the coordinated development of planning and scheduling solutions. The architecture proposed by Yung and Yang [34] is composed of functional and information agents for reducing the bull-wipe effect in a supply chain. Fox and Barbuceanu [29] have proposed an architecture for agent-based supply chain management composed of functional and information agents. They have also introduced a common building shell for the agent structure in supply chain management.

Wu et al. [15] focus on web-centric and Internet-based supply chain management. They concentrate on service delivery via collaborative agents in the Internet and propose a common and integrated framework for web-centric supply chain management systems. The EDS Group [15] applies web technology for developing a networked society for each partner in the supply chain. The group uses Java technology for internet-based purchasing and contracting.

In literature, one can hardly find any research papers or project manuscripts concentrating on uncertainty in a supply chain, specialized information distribution and/or flexibility. According to the existing uncertainty in a supply chain environment, using an approach which can address these problems seems necessary. As each partner in the supply chain has its own needs and information requirements, distributing information according to the requirement of each partner is a critical factor, little research into which has been undertaken. Achieving flexibility in the supply chain environment is one of the main concerns of the past decade. By using fuzzy agents and creating a flexible environment in the supply chain, major issues relating to coordination and collaboration can be handled and flexibility problems in the supply chain can be addressed. The main concern of this research is focused on these important issues.

ISCM MODEL

The Integrated Supply Chain Management (ISCM) system, proposed by Fox and Barbuceanu [29], encompasses a whole architecture and a general agent building shell for all agents in agent-based supply chain management. ISCM is a multi-agent approach, in which the supply chain is considered as a set of six functional and two information agents that cooperate with each other to fulfill their goals and functions. The architecture of ISCM is shown in Figure 1.

Functional agents, including logistics, order acquisition, transportation management, resource management, scheduling and dispatching, have specific functions and interact with others to achieve the supply chain goals. Information agents support functional agents to access updated information and knowledge in the supply chain. They eliminate conflicts in information resources, process the information in order to determine the most relevant content and the most appropriate form for the needs of agents, and provide periodical information for them. Information agents provide other agents a layer of shared information storage and services. Agents periodically volunteer

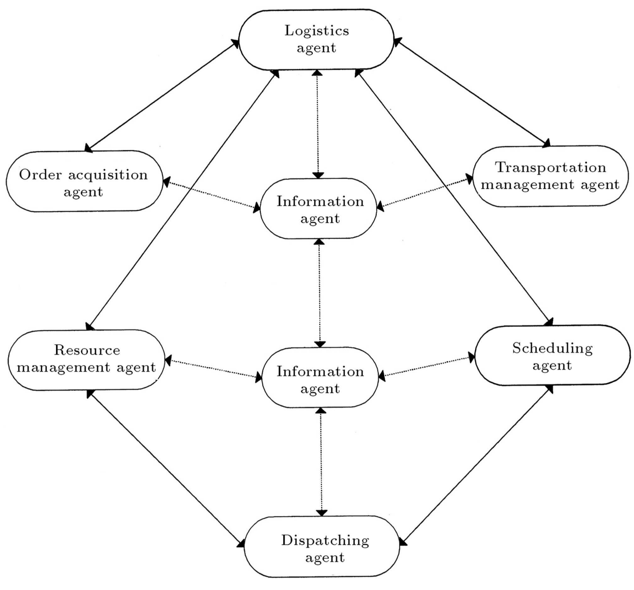

Figure 1. ISCM architecture.

some of their information to the information agents or just answer the queries sent to them by the information agent [21,29].

This paper focuses on the architecture of the information agent in ISCM. For this purpose, its functions, inputs and outputs are explained. Then, by considering the basics of a modular architecture for agents and also, supply chain properties, a new modular architecture for the information agent in ISCM is proposed. The knowledge-base in the architecture is developed and the required fuzzy rules and database are defined. Moreover, the knowledge-base is evaluated and tested and the method, in which fuzzy rules have been used is compared with the method using non-fuzzy rules. Finally, an approach for the dynamic updating of the forecasted cost and time at every stage of the supply chain is introduced.

INFORMATION AGENT ARCHITECTURE

An information agent is responsible for providing transparent access to different resources, as well as correctly retrieving, analyzing and eliminating inconsistency in data and information [35]. It is a computer software system that has access to different geographically distributed and inconsistent multi-resources and that assists users and other agents in providing relevant information. In other words, information agents manage information access issues [36]. Depending on the ability of the information agents to cooperate with each other for the execution of their tasks, they can be classified into two broad categories: Non-cooperative and cooperative. An information agent, cooperative or non-cooperative, can be relational, adaptive or mobile. Relational information agents behave and may even collaborate in order to increase their own benefits and they are utilitarian in an economic sense. Adaptive information agents are able to adapt themselves to changes in the network and information environment. Mobile information agents are able to travel autonomously through a network [35].

According to the changes in a supply chain, an agent must be able to adapt to uncertainty and with incomplete information. An approach for creating flexible behavior in an information agent is to form a team of agents which are cooperative and which are capable of gradual adaptation. Adaptive information agents can fulfill this goal. This research uses adaptive information agents to cope with changes in the supply chain environment, in order to achieve more flexibility and robustness.

The main function of an information agent is to process information retrieval requests and to monitor information intelligently and efficiently. Generally, it can be said that an information agent is able to provide essential services related to human and agent information requirements. However, there is a difference between an information agent and a web service provider. An information agent can deduce a method for analyzing requests and how they must be processed [37]. Therefore, three main functions of a typical information agent can be considered [38] as follows:

- Knowledge management,
- Eliminating conflict management,
- Supporting coordination between other agents.

According to different resources [21,25,38] and, also the supply chain environment and features, six functions have been considered necessary for an information agent in supply chain management:

- Storing of the required information for sharing and providing a layer of information,
- Analysis of information for providing the proper response to queries and requests,
- Automatic routing for information distribution,
- Conflict management,
- Change management,
- Negotiation with other agents to provide essential information.

Figure 2 demonstrates the inputs, functions and output of the proposed information agent.

The input of an information agent can be categorized as queries and changes. Requests are those that are sent by other agents and the information agent considers changes in the environment by receiving the changes. Responses to requests and queries are possible outputs of the information agent. An information agent should automatically direct the essential information to the agents. Periodical information for other agents can be another type of output. Also, an information agent should recognize which agent has access to what information. Consequently, one possible output should issue this function. Finally, an information agent must share some information between groups of agents. The output of this function can be the required shared information.

According to the above inputs, functions and outputs of the information agent in supply chain management, a new architecture is proposed for the information agent, called modular architecture [16]. For developing modular architecture, the essentials have been considered, as well as the conceptual model of an agent [38]. A conceptual model of an agent has four main parts: A reasoning engine, a knowledge-base, a learning engine and access control. A reasoning engine determines the required actions for the acquired events and knowledge from the environment. A knowledge-base stores the information and knowledge used by the

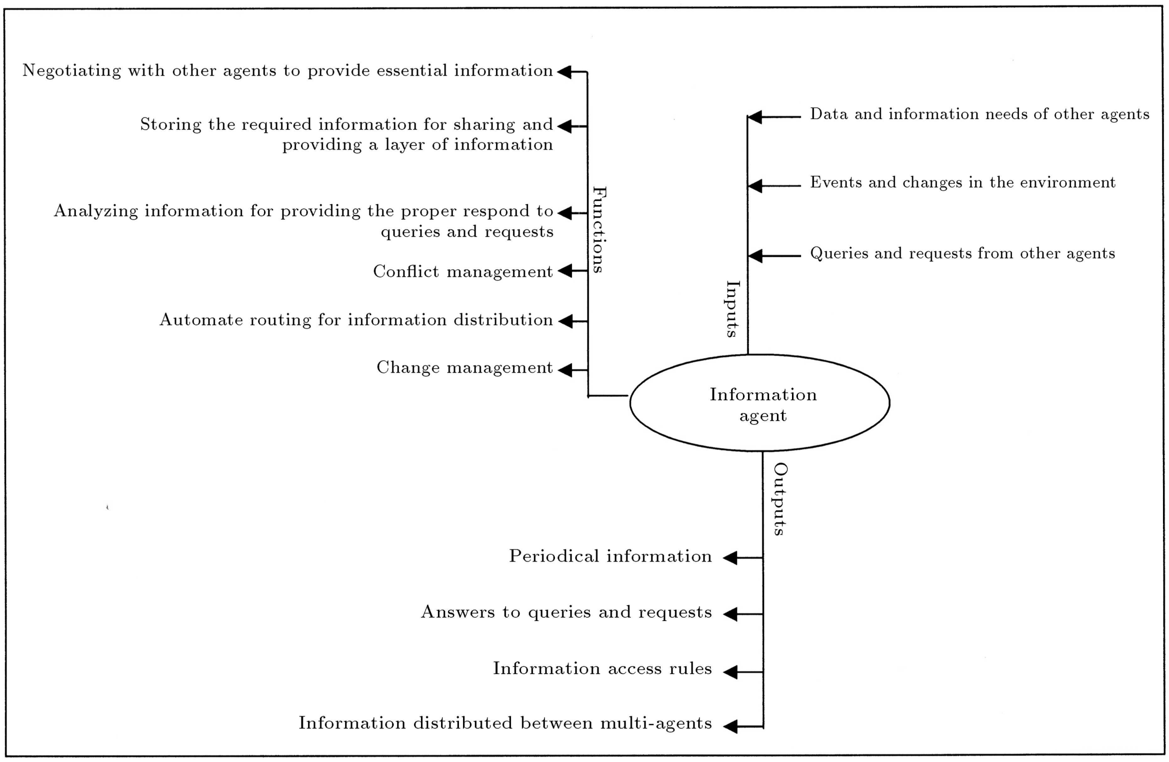

Figure 2. Inputs, functions and outputs of the information agent in supply chain management.

reasoning engine, and access control is an interface with the environment. Feedback is received by access control and actions are sent to the environment.

Considering the conceptual model of an agent, the modular architecture and, also the inputs, functions and outputs of the information agent in supply chain management, the proper architecture is proposed in Figure 3.

MODULES OF THE PROPOSED SYSTEM

This section explains the goals, features, method and structure of each module in the proposed architecture.

Conflict Management

An information agent can access different information resources and receive different kinds of data. There

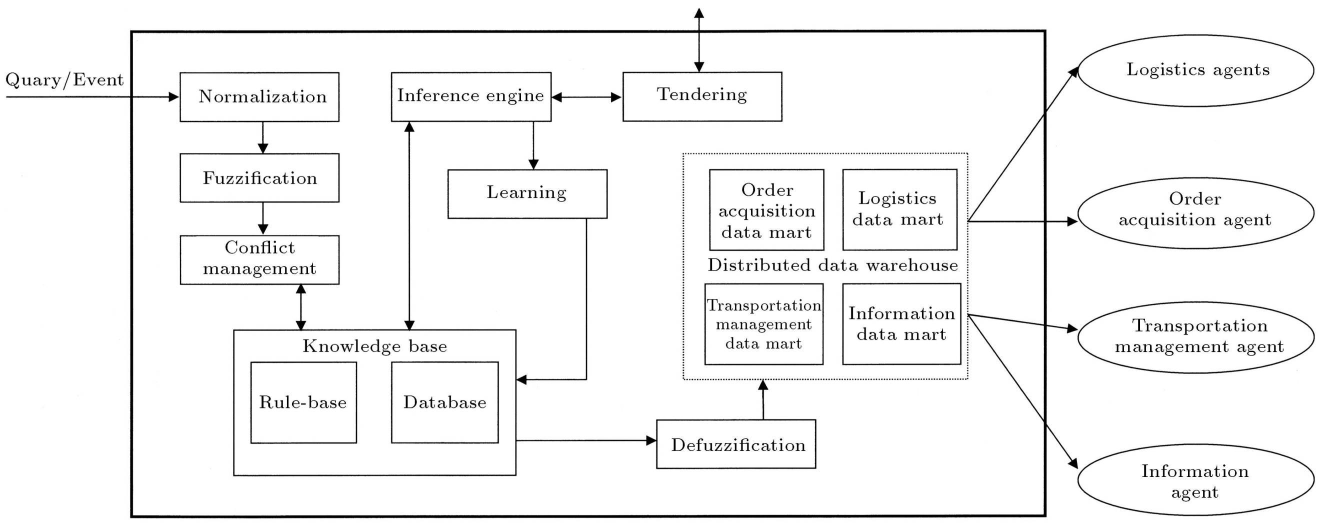

Figure 3. The proposed architecture for the information agent.

must be a module to remove any possible conflicts and inconsistencies between information. Thus, before considering any changes or information in the knowledge-base and, respectively, informing others, the conflict management module must eliminate any inconsistencies or conflict with existing information. For this purpose, an "$a - u$" space model has been used [38].

Suppose that there is a conflict between expression p and q. Expression p is the input statement and expression q is an existing statement. To each p, one can attach an authority measure (the authority of its producer) and a un-deniability measure, derived from the sum of deniability costs of all propositions that would have to be retracted if p were retracted. A high authority means that the proposition is more difficult to retract, since a high authority has to be contradicted. A high undeniability means that the proposition is more difficult to retract, because the costs of retraction incurred by consumer agents would be high [38]. These two values of all p can be represented as points in a diagram, having authority on the x-axis and undeniability on the y-axis. Such a diagram is called "$a - u$" space and is illustrated in Figure 4.

One can summarize the evaluation of a-u space in four rules as follows:

Rule 1 If $a < a_t$ AND $u < u_t$ THEN Status = No Negotiation

Rule 2 If $a < a_t$ AND $u > u_t$ THEN Status = Negotiation with Consumers

Rule 3 If $a > a_t$ AND $u < u_t$ THEN Status = Negotiation with Producer

Rule 4 If $a > a_t$ AND $u > u_t$ THEN Status = Negotiation with both.

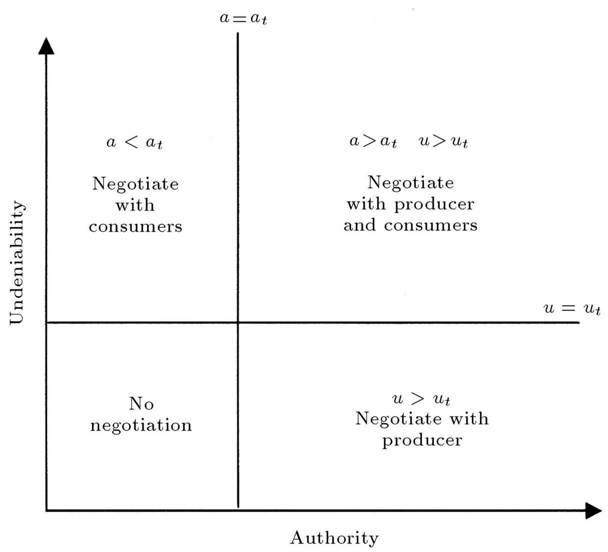

Figure 4. Negotiation regions in a-u space [40].

Using fuzzy concepts in the proposed architecture, the following fuzzy rules are presented:

Rule 1 If a isr Low AND u isr Low THEN Status is No Negotiation

Rule 2 If a isr Low AND u isr High THEN Status is Negotiation with Consumer

Rule 3 If a isr High AND u isr Low THEN Status is Negotiation with Producer

Rule 4 If a isr High AND u isr High THEN Status is Negotiation with Both.

where "High" and "Low" are linguistic values, each having its related membership function and 'isr' means 'is related to'.

Knowledge-Base

The knowledge-base is responsible for storing data and knowledge and is comprised of two parts: Rule-base and database. The rule-base contains some Meta-rules and subsets of rules, which are in both fuzzy and crisp format. The database stores data and information acquired from other external resources or new information generated by learning and tendering modules. The membership functions of the linguistic values for fuzzy rules are also stored in the database.

Inference Engine

The inference engine is one of the most important parts in an agent-based system. Reasoning and deduction process are arranged by the inference engine. Based on the situation of the inputs, the engine fires the rules in the rule-base as a matter of degree and determines the proper fuzzy output.

Tendering

As an information agent may not have the essential knowledge to provide proper answers to some queries. A tendering module has been set in the architecture to avoid leaving a query without any response. The information agent can negotiate with other agents, to provide appropriate responses to a query for which it does not have the required knowledge. Thus, it can use a tendering process to discover the response.

For organizing the tendering process, a brokering method has been used [35].

Three types of agent are differentiated from each other in brokering method as follows:

1. Provider agents: Offer their capabilities to their users and other agents;
2. Requester agents: Consume information and services offered by provider agents in the system.

Requests for provider agent capabilities have to be sent to a middle agent;

3. Middle agents, i.e. broker agents: Mediate among requesters and providers for some mutually beneficial collaboration. Each provider must first register itself with one (or multiple) middle agent(s). Provider agents advertise their capabilities (advertisement) by sending some appropriate messages describing the kind of services they offer.

The broker agent deals with the task of contracting the relevant providers, transmitting the service request to the service provider and communicating the results to the requester. Figure 5 demonstrates the method of service brokering.

When there is not enough knowledge to respond to a request or query, the information agent uses the tendering module to find the appropriate response. On this occasion, the information agent is a broker agent, the requester agent is the agent that inquires and the provider agent is the agent that provides the appropriate answer for the inquiry.

Learning

The learning ability of an agent determines the degree of its intelligence [35]. This module creates new knowledge and reduces existing errors in current knowledge. In this research, a Neural Network (NN) is implemented for learning. This net can improve the forecasted amount of either cost or time, by using an error reduction function. In this case, learning is done for long term data and, for error reduction of a short term, the NN uses the rules in the rule-base. More details of this module will be explained in the following sections. Regarding information agents, learning implies the agent's ability to automatically modify the rule-base and the facts in two ways:

- Adding new rules or modifying existing rules: if the information agent can recognize a desirable new behavior, it may be able to propose a new rule about it. Also, it can modify existing rules, which is a form of rule optimization. The tendering module handles queries for which there is no proper answer. The learning engine can add a new rule or modify existing ones in order to consider the results of tendering in the rule-base. Consequently, if the query is repeated, there is no need for tendering, because the learning engine has created the proper knowledge. Also, an information agent can recognize the periodic needs of every agent, as they request information by use of the learning engine. The learning engine can also change the certainty factors of fuzzy rules, if required.
- Adding new facts, modifying old facts, changing and improving forecasted values are a critical issue in the supply chain. The learning module can update forecasted values, e.g. costs and time, and reduce errors using fuzzy neural networks.

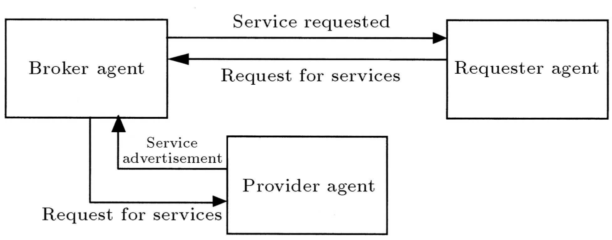

Figure 5. Service brokering method.

Distributed Data Warehouse and Data Marts

For sharing of information between other agents and allowing access to the information and data, a Distributed Data Warehouse (DDW) has been set. A DDW is a logically integrated collection of shared data that is physically distributed across the nodes of a computer network [39]. Traditional data warehouses, which are not distributed, are not appropriate for this purpose, because:

- According to the huge interaction of data, designing and developing a single traditional data warehouse is hardly possible;
- Traditional data warehouses are usually designed for predetermined requests, but here one cannot exactly determine the requests;
- Response time and loading of traditional data warehouses are much more than DDW;
- As information agents interact with different types of agent, they have to provide different kinds of information specified for each agent. Thus, in this case, a DDW composed of different data marts is appropriate;
- On occasions when the information is distributed naturally, like a supply chain, DDW can be the best solution.

A DDW is composed of different data marts, each of which is responsible for providing the information related to a specified area. A data mart is an application-focused miniature data warehouse, built rapidly to support a single line of business and share all the other characteristics of a data warehouse [20]. The data marts are independent data marts which can operate without the support of a centralized DDW. These kinds of data mart receive data and information from the resources directly, and service the consumers.

As the information agent services four different agents, including order acquisition, logistics, transportation management and the information agent, a

DDW is generated with 4 data marts. Each data mart is responsible for providing information services for each agent.

Databases and data marts in this architecture communicate with each other by the use of ORB technology. An Object Request Broker (ORB) is a middleware that establishes the client-server relationships between objects. It provides a mechanism for transparently communicating client requests to servers. ORB is an attempt to distribute computing across multiple platforms. Using an ORB, a client can transparently invoke a method on a server object, which can be on the same machine or across a network. The ORB intercepts the call and is responsible for finding an object that can implement the request, passing it to the parameters, invoking its method, and returning the results. The client does not have to be aware of where the object is located, the programming language in which it is written, the operating system on which it is running, or any other implementation details that are not part of the object's interface. Thus, the ORB provides interoperability between applications on different machines in heterogeneous distributed environments and seamlessly interconnects multiple object systems. A leading example of this approach is the Common Object Request Broker Architecture (CORBA).

Normalization, Fuzzification and Defuzzification

Mamdani type operators have been used to fuzzify the variables and aggregation of the rules stated in [40]. Mamdani fuzzy reasoning takes the minimum of the antecedent conditions in each rule and assumes the fuzzy truth of the rule to be 1. A minimum operator is used for rule implications and an AND operator in the antecedent of the rules. From a functional point of view, a Mamdani fuzzy inference system is nonlinear mapping from an input domain, $X \epsilon R^n$, to an output domain, $Y \epsilon R^m$. This input/output mapping is realized by means of R rules of the following form:

$$\text{IF } x \text{ isr } A^{(r)} \text{ THEN } y \text{ is } B^{(r)}, \tag{1}$$

where $r = 1, 2, \cdots, R$ is the index of the rule, while $A(r)$ and $B(r)$ are fuzzy relations over X and Y, respectively. When an input vector, x, is presented to the system, a fuzzy set, B, is inferred, according to the following relation:

$$B(y) = \vee(A^{(r)}(x) \wedge B^{(r)}(y)), \tag{2}$$

where, the formalism, $A(\bullet)$, denotes the membership function of a fuzzy set, A, and $\wedge$, $\vee$ are a T-norm and a T-conorm, respectively (usually the min and the max operators are used).

The center of gravity method has been used for the defuzzification method defined as:

$$\tilde{y} = \frac{\int B(y).y.dy}{\int B(y).dy}. \tag{3}$$

As the input information and data are not in the same scale, there must be a module to standardize them and make them into one form. Therefore, a normalization method is used to standardize input information and data. For normalizing a set of input information and data, the data is divided by the largest one in the set.

DEVELOPING DATABASE

As described earlier, the database contains two types of data: Data and information related to the supply chain and membership functions of the linguistic values. A relational approach has been used to develop the database. For storing the data and information related to the supply chain, a model is considered for order fulfillment, in which a supply chain is viewed as the composition of different stages. Figure 6 demonstrates the model.

As illustrated in this figure, for an order fulfillment, some certain stages must be undergone [1,2,14]. For undertaking each stage, there are different methods. Therefore, by composing different methods for each stage, there will be different routes to fulfill an order. Each method, at each stage, has three features: Method name, method cost and method time (duration). Consequently, a route for an order has three features: Route name, route cost (order total cost) and route time (order total time). The route cost and route time of an order are, respectively, the sum of the cost and time of all the stages on the route. There are two values for the time and cost of each method at every stage: The forecasted value and the actual value. Order properties and related information, such as customer properties, due date, order time and the like, are stored in the database. In addition to the mentioned information, any information agent can store every type of information received from other agents.

There are five different linguistic values for measuring the degree of changes and weights: Very Low, Low, Medium, High and Very High. The membership function for each value is shown in Figure 7.

DEVELOPING RULE-BASE

The rule-base in the knowledge-base has 31 rules, 3 Meta-rules and 3 rule subsets. The rule-base contains both fuzzy and crisp rules. Figure 8 shows the hierarchical view of the rules.

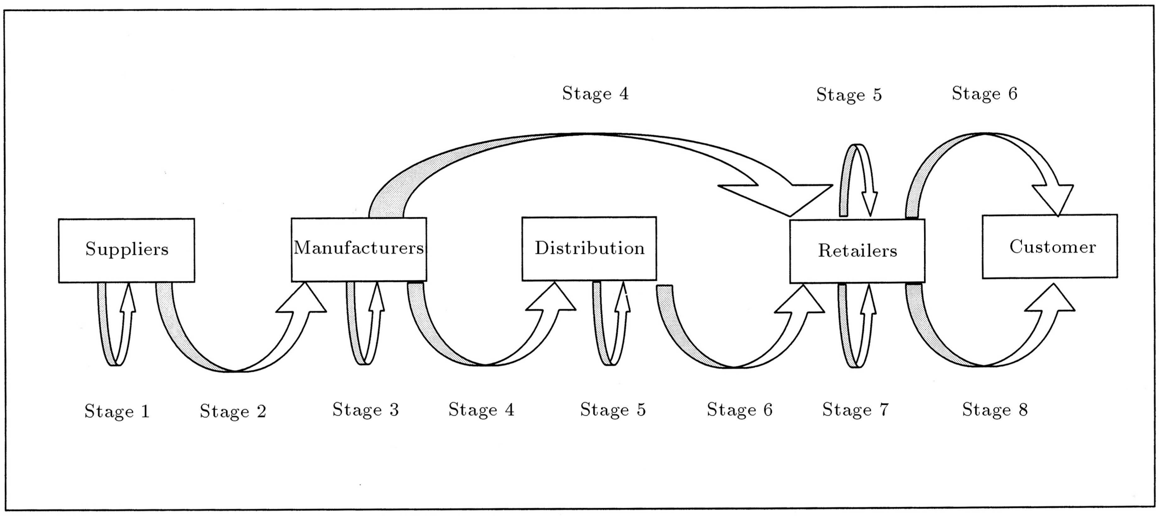

Figure 6. A model for order fulfillment.

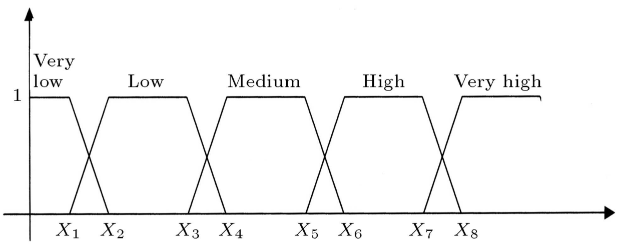

Figure 7. Membership functions of linguistic values.

DYNAMIC UPDATING BY FUZZY RULES

As mentioned before, every order has its related total time and total cost, which can be obtained, respectively, by adding the time and cost of each stage on the order route. An approach is introduced for updating the forecasted values of the total time and cost of each order and for directing the supply chain to the committed cost and time for the customer. Before an order flows in the supply chain, there is a forecasted total time and cost for the order fulfillment. As the order moves in the chain and takes the stages, the actual cost and time for each stage are emerged. Consequently, a new total time and cost for the order can be defined. Thus, if order a needs m stages to be fulfilled, one has:

Total time for:

$$a : \text{TotalTime } (a) = \sum_{i=1}^{m} \text{TimeStage}_i(a). \tag{4}$$

Total cost for:

$$a : \text{TotalCost } (a) = \sum_{i=1}^{m} \text{CostStage}_i(a), \tag{5}$$

where, $\text{TimeStage}_i(a)$ is the duration of stage i and $\text{CostStage}_i(a)$is the cost of stage i of order a. If the time of stage k changes to T, then, the total time of a changes to:

New total time of:

$$a : \text{TotalTime } (a) = \sum_{i=1}^{m} \text{TimeStage}_i(a) + (T - \text{TimeStage}_k(a)). \tag{6}$$

Generally, at stage i of m, one has i total time and i total cost for a. Thus, one has m total time and m total cost for order a at the end of the order fulfillment. A different total time and total cost have been used at every stage to forecast the next value in the chain. By comparing the forecasted value at every stage and the initial forecast value, which is the committed value to the costumer, the partners in the chain will be notified regarding the difference between the committed and the real value and, consequently, they can change their plans and behavior in such a manner as to reach the committed value.

DETERMINATION OF THE FORECASTED VALUE

Assume that one is at stage i of order a. Thus, one has i total time as follows:

$$\left\{\begin{array}{ll} \text{Total time of } a \text{ in stage } 1 : & \text{TotalTime}^{(1)}(a) = T_1 \\ \text{Total time of } a \text{ in stage } 2 : & \text{TotalTime}^{(2)}(a) = T_2 \\ \vdots & \vdots \\ \text{Total time of } a \text{ in stage } i : & \text{TotalTime}^{(i)}(a) = T_i \end{array}\right\}. \tag{7}$$

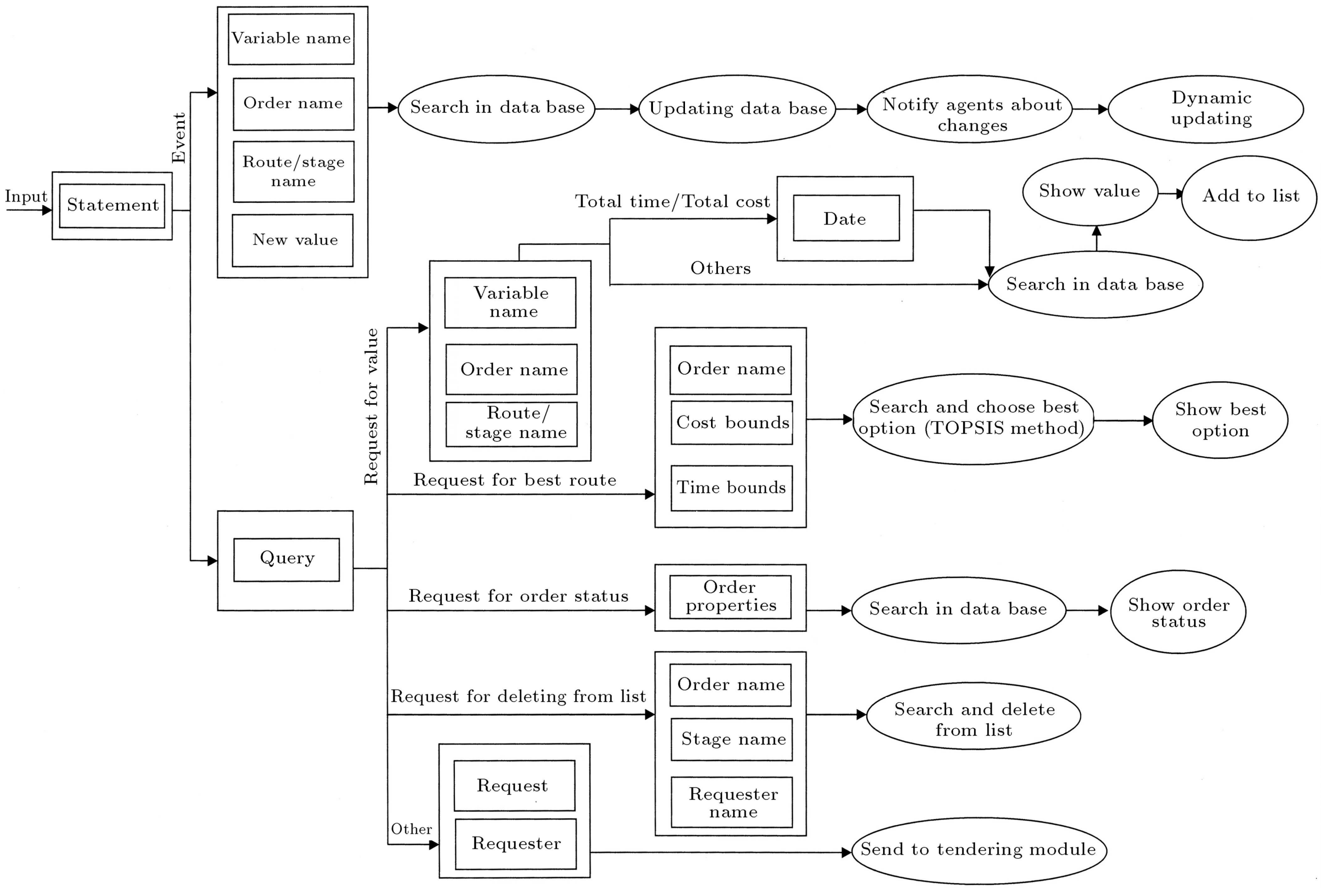

Figure 8. Hierarchical view of rules.

Then, one can use fuzzy rules to define a weight for each total time, for example:

$$\text{IF } (T_k - T_{k-1}) \text{ isr Low AND } (T_{k-1} - T_{k-2}) \text{ isr High THEN Weight}(T_k) \text{ isr Medium,}$$

where, Weight(T_k) is the weight of T_k and "isr" means "is related to". The forecasted value for total time at stage i is:

$$T = \sum_{k=1}^{i} T_k \text{ Weight}(T_k). \tag{8}$$

TECHNICAL ISSUES, IMPLEMENTATION AND VALIDATION

The proposed common structure for agents in a supply chain by Fox and Barbuceanu [29] is composed of seven layers, which are: Agent communication language, conversational coordination, action and behavior, organization model, decision making, behavior planning, behavior execution and domain specific solvers. The structure is considered for all agents in agent-based supply chain management, including information and functional agents. They also proposed some essential features for an information agent as follows [38]:

- Change management,
- Query management,
- Conflict management,
- Time map management system.

The proposed architecture in this paper has eight modules. These modules satisfy the expected functions and output from the information agent. The normalization module standardizes the input data and changes their format in such a manner that their comparison makes sense. The fuzzification module fuzzifies the information to deduce from and use in the knowledge-base. Using fuzzy rules increases system flexibility and robustness and reduces complexity, thus, the information agent can handle uncertain events and information more easily. Conflict management eliminates existing conflicts and inconsistencies between data and information by using the $a-u$ space model with fuzzy boundaries. The knowledge-base stores current and new knowledge, including rules and data. The tendering module avoids leaving a query without any response and the information agent negotiates with other agents to find the proper answer for any query that it cannot respond to individually. The learning module modifies existing rules and facts and eventually creates new ones, according to current information, events and the agents' behavior. For sharing information and data, specifically for every agent, a DDW module has been applied. The DDW contains four data marts, each responsible for providing specialized information and services for a particular agent. The defuzzification module changes fuzzy information to crisp information to share with other agents. For testing the rule-base and comparing the results of applying fuzzy rules, a set of data related to a Computer Monitor supply chain has been used. All the proposed rules and their related databases have been assessed and suitable responses have been gained from their assessment, based on negotiations with experts. For the development of the database, Microsoft Access XP was used and for the development of the knowledge-base, Visual Basic, together with MATLAB, was used. As mentioned before, the dynamic updating rules are considered to be Fuzzy. In Fuzzy Expert Systems, all rules will be fired as a matter of degree.

Figure 9 shows the results of dynamic updating and the behavior of the chain. In this figure, the actual total cost of every stage and its corresponding forecasted value (fuzzy method) are illustrated. In fact, by using a dynamic updating approach, the forecasted values at every stage are determined. This value has an important role in following further stages in the supply chain. If the forecasted value at a stage is different from the committed value, i.e. the initial forecasted value, the partners in the supply chain can adapt themselves to the change and behave in such a manner as to decrease the effect of changes at the next stages. This adaptation is repeated at each stage and if, for any stage, a non-predictable event occurs, where the total cost goes beyond the committed value, the forecasted value at that stage advises the chain to avoid these kinds of change at the next stages. As illustrated in Figure 8, the forecasted total cost controls actual values and directs them to achieve the committed value at the end of the supply chain.

To compare the dynamic updating fuzzy rules with non-fuzzy ones, both predicted and updated results are investigated. In the first experiment, dynamic updating with fuzzy rules has been used. In the second experiment, where the dynamic updating with non-fuzzy rules has been used, a simple average has been implemented without any weights. This comparison has been undertaken for certain total order costs in its related time period. The results of the predicted amount, using a fuzzy approach, and the real amount for the total order cost is shown in Figure 9. In Figure 10, the results of using fuzzy prediction and the average (mean) approach is demonstrated. As a matter of fact, the predicted amount at each stage is determined by using dynamic updating. The prediction amount plays an important role in the next stages of the order investigation. So, when the predicted total cost at a stage is different from the related customer order, the supply chain adjusts itself to compensate this difference, in such a way as to satisfy the customers. This adjustment is done dynamically and iteratively to minimize the change in the promised amount. As shown in Figure 9, the total cost of the promised amount is \$252. In this figure, it can be observed that, at the last stages of each order research, the real amount is close to the predicted amount and, so, is closer to the amount promised to the customer.

Traditionally, a simple method for computing the forecasted value is to calculate the average of the actual value at every stage, without considering any weight for them. For obtaining the new forecasted value, fuzzy logic has been implemented and a weight has been considered for each actual value, at every stage,

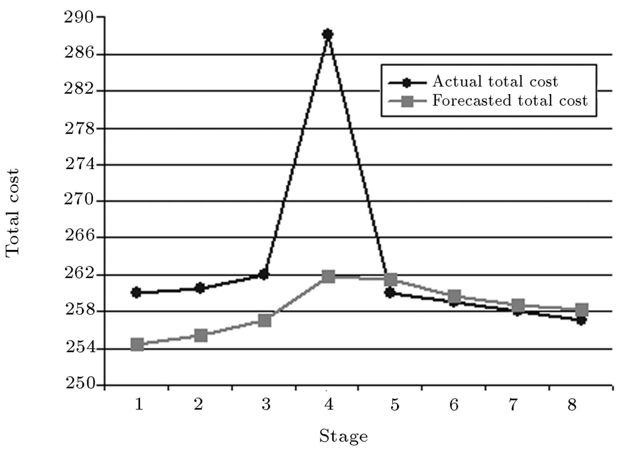

Figure 9. Actual and forecasted total cost of an order (fuzzy method).

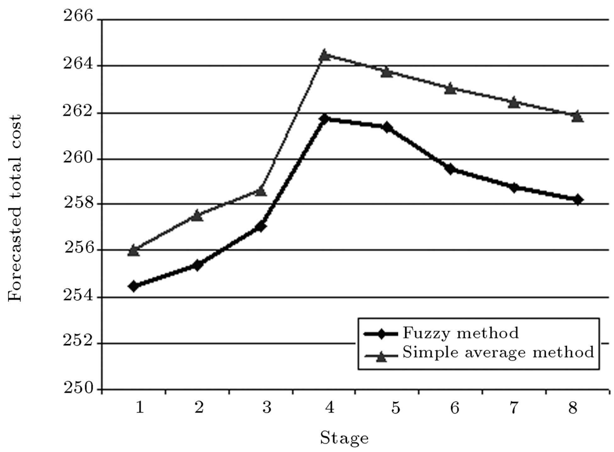

Figure 10. Forecasted total cost by fuzzy and simple average method.

using fuzzy rules. This helps to reduce the effect of unexpected events and, therefore, produces more realistic forecasted values. In Figure 10, two methods for calculating the forecasted value at every stage; a simple average and a fuzzy method for the total cost of an order, are shown.

As stated above, the forecasted amount, using fuzzy rules and dynamic updating, is more realistic than using a mean approach, leading to error reduction. This is mainly due to the fact that, in the fuzzy approach, each item has its related weight, based on the degree of importance. Obviously, in the SC environment, drastic changes are encountered. In this environment, if one uses identity weights, the forecasting amount might not be close to the real one. Thus, the forecasted results not only mislead the SC in reaching its real goals, but also prevent it from providing proper planning. On the other hand, using fuzzy rules can strengthen and increase the robustness and flexibility of the system, since using crisp rules needs crisp numbers in each increasing and decreasing of costs. In such cases, the number of variables and their related values will increase, which leads to more complexity and prolongs response times. Furthermore, using fuzzy logic leads to increasing the adaptability of the system to adjust itself with real world problems. In such a case, while each variable changes, again, the fuzzy rules are interpretable. The existence of this capability for information agents increases the adaptability of the system with environmental changes and, so, increases transformation of the event to the agents, in order to inform them.

As shown in this figure, the fuzzy method is closer to the committed value (252) than the simple average method. Therefore, the fuzzy forecasted value is more effective, in directing the actual values to reach the committed value, than the simple average method.

In Figure 10, fuzzy forecasting and the average approach are compared. As shown in this figure, the fuzzy methodology is very close to the promised amount, i.e. $252. So, the capability of the fuzzy forecasting approach in cooperation and the leading of the supply chain elements in the direction of customer satisfaction and in reaching the main goals of the SC (such as minimization of order cost and time and, as a result, customer satisfaction), is better than in the average approach.

CONCLUSIONS

This paper proposes a proper modular architecture for the information agent, based on the inputs, functions and outputs of the agent. The proposed architecture has nine different modules, each of which is responsible for one or more function(s) for the information agent. The fuzzification module in the architecture fuzzifies data and information. As a result, an information agent can handle changes in a flexible manner. Also, a tendering module has been proposed, by which the information agent can find the answer to any query and/or request from other agents. The tendering module uses a brokering approach to handle its responsibilities. The conflict management module handles inconsistencies in data and information and eliminates them by the use of fuzzy boundaries. By the use of knowledge-base and inference modules, query answers and information requirements for other agents in the environment are determined. Also, a learning module is considered to generate new data and information and change existing rules, if necessary. A Distributed Data Warehouse (DDW) module has been considered to address data and information sharing between agents and automatically distribute specialized information for each agent.

Moreover, the knowledge-base module, which is composed of a rule-base and a database, has been developed. For updating forecasted cost and time values in the supply chain for an order, a new approach, called dynamic updating, has been implemented. By using dynamic updating, a new forecasted value can be generated at each stage of an order fulfillment. This approach directs the supply chain partners to fulfill commitments to the end-customer and achieve minimum deviation from commitments at the end of the chain. The approach is based on fuzzy rules. Finally, the results of testing the knowledge-base have been addressed. The results show that using the dynamic updating approach forces the supply chain partner to consider time and cost commitments for customers during order fulfillment. This approach makes supply chain partners deliver the order in a timely and cost effective manner. In addition, using fuzzy concepts (fuzzy methods) in calculating the forecasted value in the dynamic updating approach, improves the forecasted values at each stage of an order fulfillment. Using fuzzy rules, a weight has been determined for each forecasted cost and time value. A weight for each value shows its effect and role in calculating new forecasted values for each stage.

This work has potential for further research. Future research into expanding tendering, learning and the DDW modules of the proposed architecture is required. Also, a proper architecture for other agents in the ISCM model can be determined, according to their functions, inputs and outputs, especially for the other information agent in the model, who interacts with the dispatching, scheduling and resource management agent. Moreover, in this research, trapezoidal membership functions have been used. In future research, an optimization method to generate the membership functions will be presented, based on the behavior of the system, automatically. Finally, this research has used the Mamdani type of approximate reasoning in its inference engine. Other parametric fuzzy reasoning, such as a combination of FATI and FITA, may make the engine more robust.

REFERENCES

1. Handfield, R.B. and Nichols E.L., *An Introduction to Supply Chain Management*, Prentice Hall (1998).
2. Rogers, D., *Introduction to Supply Chain Management*, Center for Logistics Management, University of Nevada (2003).
3. Graves, S.C. and Willems, S.P. "Optimizing strategic safety stock placement in supply chains", *Manufacturing and Service Operation Management*, **2**(1), pp 68-83 (2000).
4. Graves, S.C. and Willems, S.P. "Optimizing the supply chain configuration for new products", *Proceeding of the 2000 MSOM Conference*, Ann Arbor, MI (2000).
5. Cebi, F. and Bayraktar, D. "An integrated approach for supplier selection", *Logistics Information Management*, **16**(6), pp 395-400 (2003).
6. Wang, G., Hang, S.H. and Dismukes, J.P. "Product-driven supply chain selection using multi-criteria decision making methodology", *International Journal of Production Economics*, **91**, pp 1-15 (2004).
7. Stadtler, M. "Supply chain management and advanced planning-basics, overview and challenge", *European Journal of Operational Research*, **163**, pp 575-588 (2005).
8. Baganha, M. and Cohen, M. "The stabilizing effect of inventory in supply chains", *Operations Research*, **46**, pp 572-83 (1998).
9. Graves, S.C. "A single-item inventory model for a non-stationary demand process", *Manufacturing & Service Operations Management*, **1**, pp 50-61 (1999).
10. Chen, F., Drezner, Z., Ryan, J. and Simchi-Levi, D. "Quantifying the bullwhip effect in a simple supply chain: The impact of forecasting, lead-times, and information", *Management Science*, **46**(3), pp 436-43 (2000).
11. Li, G., Wang, S., Yan, H. and Yu, G. "Information transformation in a supply chain", *Computers and Operations Research*, **32**, pp 707-725 (2005).
12. Lambert, M.D., Cooper, M. and Pugh, J.D. "Supply chain management implementation issues and research opportunities", *The International Journal of Logistics Management*, **9**(2), pp 1-19 (1998).
13. Verdicchio, M. and Colombetti, M. "Commitments for agent-based supply chain management", *ACM SIGecom Exchange*, **3**(1), p 11 (2000).
14. Simchi-Levi, D., Kaminsky, P. and Simchi-Levi, E., *Designing and Managing the Supply Chain*, McGraw-Hill (2000).
15. Wu, J., Ulieru, M., Cobzaru, M. and Norrie, D.H., *Agent-Based Supply Chain Management Systems: State of the Art and Implementation Issues*, University of Calgary (2000).
16. Shen, W., Norrie, D.H. and Barthes, J.-P., *Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing*, Taylor & Francis (2001).
17. Fazlollahi, B., *Soft Computing Agent-Based Distributed Intelligent Systems*, IOS Press (2002).
18. Erol, E. and Ferrel, W.G. "A method for selection problems with multiple conflicting objectives and both qualitative and quantitative criteria", *International Journal of Production Economics*, **86**, pp 187-199 (2003).
19. Fazel Zarandi, M.H. and Saghiri S., "Developing fuzzy expert systems models for supply chain complex problem: A comparison with linear programming", *IEEE World Congress on Computational Intelligence*, Vancouver, Canada, pp 6935-6939 (2006).
20. Zarandi, M.H.F., Turksen, I.B. and Saghiri, S. "Fuzzy multiple objective supplier selection in multiple products and supplier environment", *International Journal of Engineering Science*, **16**(2), pp 1-20 (2005).
21. Fox, M.S., Chionglo, J.F. and Barbuceanu, M., *The Integrated Supply Chain Management System*, University of Toronto (1993).
22. Hayzelden, A.L.G. and Bourne, R.A., *Agent Technology for Communication Infrastructures*, John Willey & Sons (2001).
23. Liu, D.Y., Yang, K. and Chen, J.Z. "Agents: Present status and trends", *Journal of Software*, **11**(3), pp 315-321 (2000).
24. Nwana, H.S. and Ndumu, D.T. "A perspective on software agents research", *Knowledge Engineering Review*, **14**(2), pp 125-142 (1999).
25. Sycara, K., *In-Context Information Management Through Adaptive Collaboration of Intelligent Agents*, published in Intelligent Information Agents, Springer (1999).
26. Jiao, J.R., You, X. and Kumar, A. "An agent-based framework for collaborative negotiation in the global manufacturing supply chain network", *Robotics and Computer-Integrated Manufacturing*, **22**, pp 239-255 (2006).
27. Xue, X., Li, X., Shen, Q. and Wang, Y. "An agent-based framework for supply chain coordination in construction", *Journal of Automation in Construction*, **14**, pp 413-430 (2005).
28. Wang, Y. and Sang, D. "Multi-agent framework for third party logistics in E-commerce", *Expert Systems with Applications*, **29**, pp 431-436 (2005).
29. Fox, M.S. and Barbuceanu, M., *Agent Oriented Supply Chain Management*, Kluwer (2000).
30. Dasgupta, P., Narasimhan, N., Moser, L. and Melliar-Smith, P. "MAgNET: Mobile agents for networked electronic trading", *IEEE Transactions on Knowledge and Data Engineering*, **11**(4), pp 509-525 (1999).

31. Chauhan, D. "JAFMAS: A java-based agent framework for multi-agent systems development and implementation", *Technical Report*, ECECS Department, University of Cincinnati (1997).

32. Lau, H.C.W., Ning, A., Chan, F.T.S. and Ip, R.W.L. "Design and development of a flexible workflow supply chain system", *IEEE*, **2**, pp 792-796 (2000).

33. Ulieru, M., Norrie, D.H. and Shen, W. "A multi-agent system for supply chain management", *Progress in Simulation, Modeling, Analysis and Synthesis of Modern Electrical and Electronic Devices and Systems*, World Scientific and Engineering Society Press, pp 342-348 (1999).

34. Yung, S.K. and Yang, C.C. "A new approach to solve supply chain management problem by integrating multi-agent technology and constraint network", in *Proceedings of the Thirty-Second Annual Hawaii International Conference on System Sciences* (1998).

35. Klusch, M., *Intelligent Information Agents*, Springer (1999).

36. Nwana, H.S. "Software agents: An overview", *Knowledge Engineering Review*, **11**(3), pp 1-40 (1996).

37. Caglayan, A. and Harrison, C., *Agent Sourcebook*, John Wiley & Sons (1997).

38. Barbuceabu, M. and Fox, M.S. "The information agent: An infrastructure agent supporting collaborative enterprise architectures", *Report of Enterprise Integration*, University of Toronto (1995).

39. Moeller, R.A. "Distributed data warehousing using web technologies", *American Management Association; AMACOM*, New York, USA (2001).

40. Cordon, O., Herrera, F. and Hoffman, F. "Genetic fuzzy systems: Evolutionary tuning of learning of fuzzy knowledge-bases", *Advanced in Fuzzy Systems-Applications and Theory*, **19**, *World Scientific* (2001).