AGENT-BASED INTELLIGENT COLLABORATIVE MECHANISM

Adina-Georgeta CRETAN[*]

Abstract

This paper proposes a collaborative intelligent mechanism to support concurrent negotiations among organizations acting in the same industrial market. Each organization has limited resources and in order to better accomplish a higher external demand, the managers are forced to outsource parts of their contracts even to concurrent organizations. In this concurrent environment each organization wants to preserve its decision autonomy and to disclose as little as possible from its business information. The complexity of our negotiation model is done by the dynamic environment in which multi-attribute and multi-participant negotiations are racing over the same set of resources. We are using the metaphor Interaction Abstract Machines (IAMs) to model the parallelism and the non-deterministic aspects of our negotiation process.

Keywords: *Negotiation model, web services, collaborative mechanism, dynamic environment, multi-agent systems.*

1. Introduction

The advent of the Internet and more recently the cloud-computing trend have led to the development of various forms of virtual collaboration in which the organizations are trying to exploit the facilities of the network to achieve higher utilization of their resources. We try to provide support to these collaboration activities and we propose negotiation as a fundamental mechanism for such collaborations.

The concept of "Virtual Enterprise (VE)" or "Network of Enterprises" has emerged to identify the situation when several independent companies decided to collaborate and establish a virtual organization with the goal of increasing their profits. Camarinha-Matos defines the concept of VE as follows: "A *Virtual Enterprise (VE)* is a temporary alliance of enterprises that come together to share skills and resources in order to better respond to business opportunities and whose cooperation is supported by computer networks"[1].

In this paper we present how organizations participate and control the status of the negotiations and how the negotiation processes are managed.

The starting point in the development of this work was the goal to support small and medium enterprises that are not able or are not willing to perform alone a large contract since in this situation the association in a virtual alliance provides the opportunity to subcontract the tasks of the contract to other partners within the alliance. To achieve this goal, research was dedicated to the development of a model to coordinate the negotiations that take place within an inter-organizational alliance. Our research was focused on the topics of virtual alliances, automation of the negotiations and of coordination aimed to provide the mechanisms for coordinating the negotiations that take place among autonomous enterprises that are grouped in a virtual alliance.

Assuming that the nature of the roles that may be played in a negotiation are similar in multiple approaches, the number of participants involved at the same time in the same negotiation is considerably different.

Depending on the number of participants involved in a negotiation, we may distinguish various negotiation types: *bilateral negotiation (one-to-one)*; *one-to-many negotiation; many-to-many negotiation*.

Taking into account the complexity of the negotiations modeled by multi-agent system, we can state that to conduct in an efficient fashion one or many negotiations that involve a large number of participants and to properly account for all negotiation dimensions, it is necessary to develop a coordination process that is defined outside of the specific constraints of a given decision mechanism or communication protocol.

The negotiation process was exemplified by scenarios tight together by a virtual alliance of the autonomous gas stations. Typically, these are competing companies. However, to satisfy the demands that go beyond the vicinity of a single gas station and to better accommodate the market requirements, they must enter in an alliance and must cooperate to achieve common tasks. The type of alliance that we use to define their association emphasizes that each participant to this alliance is completely autonomous i.e., it is responsible of its own amount of work and the management of its resources. The manager of a gas station wants to have a complete decision-making power over the administration of his contracts, resources, budget and clients. At the same time, the manager attempts to cooperate with other gas stations to accomplish the global task at hand only through a minimal exchange of information. This exchange is minimal in the sense that the manager is in charge and has the ability to select the information exchanged.

[*] Associate Professor, PhD, Faculty of Economics and Business Administration"Nicolae Titulescu" University, Bucharest (e-mail: adinacretan@univnt.ro)

[1] Camarinha-Matos L.M. and Afsarmanesh H.,(2004), *Collaborative Networked Organizations*, Kluwer Academic Publisher Boston

When a purchasing request reaches a gas station, the manager analyses it to understand if it can be accepted, taking into account job schedules and resources availability. If the manager accepts the purchasing request, he may decide to perform the job locally or to partially subcontract it, given the gas station resource availability and technical capabilities. If the manager decides to subcontract a job, he starts a negotiation within the collaborative infrastructure with selected participants. ***In case*** that the negotiation results in an agreement, a contract is settled between the subcontractor and the contractor gas station, which defines the business process ***outsourcing*** jobs and a set of obligation relations among participants[2].

The gas station alliance scenario shows a typical example of the SME virtual alliances where partner organizations may be in competition with each other, but may want to cooperate in order to be globally more responsive to market demand.

The collaborative infrastructure, that we describe, should flexibly support negotiation processes respecting the autonomy of the partners.

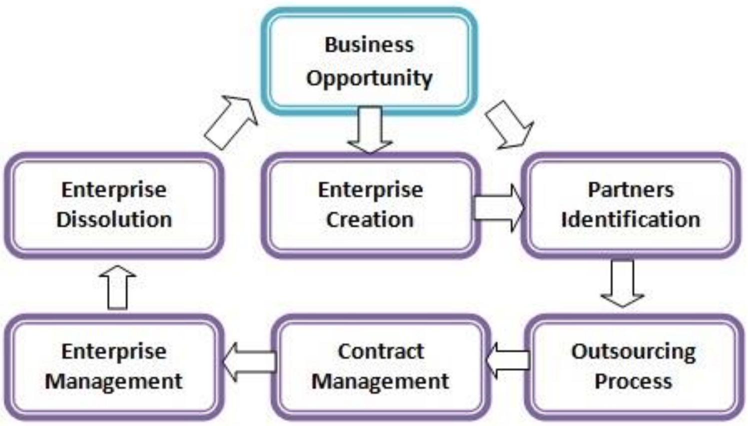
We are starting with a presentation in Section 2 of a VE life cycle model. Section 3 presents a formal interaction model to manage multiple concurrent negotiations by using the metaphor Interaction Abstract Machines (IAMs). Then, we are briefly describing in Section 4 the collaborative negotiation architecture.

The main objective of this paper is to propose an intelligent collaborative mechanism in a dynamical system with autonomous organizations. In Section 5 we define the Coordination Components that manage different negotiations which may take place simultaneously. Finally, Section 6 concludes this paper.

2. The main steps of the Virtual Enterprise life cycle

The life cycle of virtual enterprise is classified into six phases. The relevance in different phases is shown in Figure 1 and the statement for each phase is given as follows:

Fig. 1. Life-cycle of a virtual enterprise



a) VE creation

When a business opportunity is detected, there is a need to plan and create the VE, identify partners, establish the contract or cooperation agreement among partners, in order to manage the processes of the VE.

b) Partners search and selection

The selection of business partners is a very important and critical activity in the operation of a company. Partners search can be based on a number of different information sources, being private, public, or independent. The enterprise's private suppliers' list is a data repository that contains information about the companies that have had commercial relationships with this enterprise. This information composes an *Internal Suppliers Directory (ISD)*. External sources include directories maintained by industrial associations, commerce chambers, or Internet services. This information composes the *External Suppliers Directory (ESD)*. Another emerging solution is the creation of clusters of enterprises that agreed to cooperate and whose skills and available resources are registered in a common *SME Cluster Directory* (CD).

c) Outsourcing of tasks within a VE

In this stage of a VE life cycle, we can assume that a gas station company receives a customer demand. In this respect, the Manager of this company may negotiate the outsourcing of a schedule tasks that cannot perform locally with multiple partners of selected gas station companies, geographically distributed. The Manager can select the partners of the negotiation among the database possible partners according to their declared resources and the knowledge he has about them.

The outcome of a negotiation can be "success" (the task was fully outsourced), "failure" (no

[2] Singh M.P., (1997) *Commitments among autonomous agents in information-rich environments*. In Proceedings of the 8th European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW), pp. 141–155

outsourcing agreement could be reached) or "partial" (only part of the task could be outsourced).

d) Contract management in the VE

In case the negotiation process ends in a successful, a contract is established between the outsourcing company and the insourcing ones. The contract is a complex object, which is based of trust in this coordination mechanism. Moreover, it contains a set of specific rules, such as penalties, expressing obligation relations between the participants.

In case of failure of a partner, the Manager will have to supervise if the obligations are honored (for example to oblige the partner to finish his work or to set penalties) and to modify the business process renegotiating parts of the work that have not been realized.

e) Management of the VE

A VE is a dynamic entity in which a new company may join or leave it. Members may need to leave for many reasons, when they change their activity or when they don't want any more to collaborate with the partners of the VE. In case of departure from the VE, the leaving partner may either notify all the partners. It also may leave without giving any information. The departure of a partner from the VE will have an important impact on ongoing contracts especially when this partner is an insourcer of an important amount of task.

f) VE dissolution - after stopping the execution of the business processes.

3. Building the Negotiation Model

In this section we propose a formal model to settle and to manage the coordination rules of one or more negotiations, which can take place in parallel. We will introduce the metaphor of Interaction Abstract Machines (IAMs) to describe the negotiation model. We introduce the Program Formula to define the methods used to manage the parallel evolution of multiple negotiations.

3.1. The Metaphor Interaction Abstract Machines (IAMs)

The metaphor Interaction Abstract Machines (IAMs) will be used to facilitate modeling of the evolution of a *multi-attribute*, *multi-participant*, *multi-phase negotiation*. In IAMs, a system consists of different *entities* and each entity is characterized by a state that is represented as a set of *resources* [4]. It may evolve according to different laws of the following form, also called "*methods*":

A1@...@An <>- B1@...@Bm

A method is executed if the state of the entity contains all resources from the left side (called the "*head*") and, in this case, the entity may perform a transition to a new state where the old resources *(A1,…,An)* are replaced by the resources *(B1,…,Bm)* on the right side (called the "*body*"). All other resources of the entity that do not participate in the execution of the method are present in the new state.

The operators used in a method are:

- the operator @ assembles together resources that are present in the same state of an entity;
- the operator <>- indicates the transition to a new state of an entity;
- the operator & is used in the body of a method to connect several sets of resources;
- the symbol "T" is used to indicate an empty body.

In IAMs, an entity has the following characteristics:

- if there are two methods whose heads consist of two sets of distinct resources, then the methods may be executed in parallel;
- if two methods share common resources, then a single method may be executed and the selection procedure is made in a non-deterministic manner.

In IAMs, the methods may model four types of transition that may occur to an entity: *transformation*, *cloning*, *destruction* and *communication*. Through the methods of type *transformation* the state of an entity is simply transformed in a new state. If the state of the entity contains all the resources of the head of a transformation method, the entity performs a transition to a new state where the head resources are replaced by the body resources of the method. Through the methods of type *cloning* an entity is cloned in a finite number of entities that have the same state. If the state of the entity contains all the resources of a head of a cloning method and if the body of the method contains several sets of distinct resources, then the entity is cloned several times, as determined by the number of distinct sets, and each of the resulting clones suffers a transformation by replacing the head of the method with the corresponding body. In the case of a *destruction* of the state, the entity disappears. If the state of the entity contains all the resources of the head of a transformation method and, if the body of the method is the resource T, then the entity disappears.

In IAMs, the *communication* among various entities is of type broadcasting and it is represented by the symbol "^". This symbol is used to the heads of the methods to predefine the resources involved in the broadcasting. These resources are inserted in the current entity and broadcasted to all the entities existent in the system, with the exception of the current entity. This mechanism of communication thus executes two synchronous operations:

- *transformation*: if all resources that are not pre-defined at the head of the method enter in collision, then the pre-defined resources are inserted in the entity and are immediately consumed through the application of the method;
- *communication:* insertion of the copies of the pre-defined resources in all entities that are present in the system at that time instance.

3.2. Modelling the Negotiation Process

According to our approach regarding the negotiation, the participants to a negotiation may *propose* offers and each participant may decide in an autonomous manner to stop a negotiation either by *accepting* or by *rejecting* the offer received. Also, depending on its role in a negotiation, a participant may *invite* new participants to the negotiation. To model this type of negotiation, we will make use of the previously defined particles and we will propose the methods to manage the evolution of these particles.

As we have seen, a characteristic of negotiation is its multi-node image, which allows parallel development of several phases of negotiation. A possibility to continue a negotiation is to create a new phase of negotiation from an existing one. In this regard, the Figure 2 presents the possible evolutions of a ph0 phase of negotiation described by the *atom (s,ph0)*.

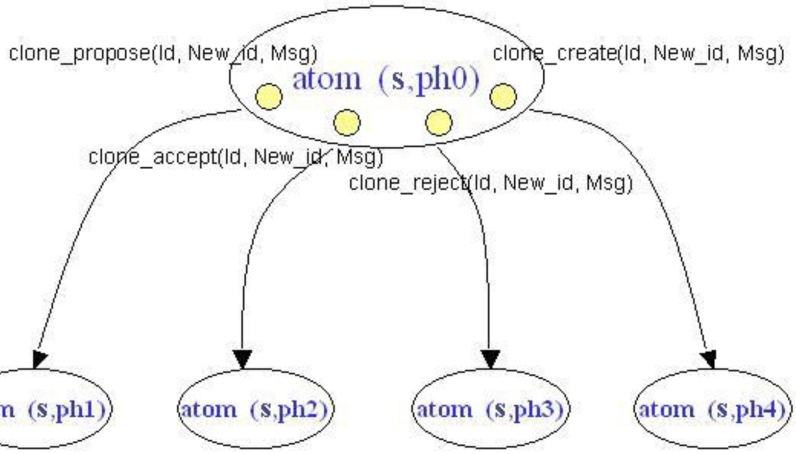


Fig. 2. Evolution of negotiation process by cloning an atom

In accordance with the aspects of negotiation for which changes are made, three new negotiation phases are possible:

- evolution of negotiated attributes and / or of their value from *atom(s,ph0)* to *atom(s,ph1)*: a participant sends a new proposal thus achieving either the *contraction* of the negotiation attributes, or their *extension*, by the introduction of new attributes to negotiate;
- evolution of the negotiation status perceived by one of the sequences sharing the new negotiation phase: one of the participants accepts - *atom(s,ph2)* - or refuses a proposal - *atom(s,ph3);*
- evolution of participants and of dependences among negotiations by the evolution of the number of sequences sharing the same negotiation phase: a sequence can invite a new sequence to share a new phase of negotiation *atom(s,ph4)*.

Through the use of the metaphor IAMs, the evolutions of the negotiation phases correspond to the evolutions at the atoms level. The evolution may be regarded as a process consisting of two stages: a *cloning* operation of the atom existent in the initial stage and a *transformation* operation within the cloned atom to allow for the new negotiation phase.

The *cloning* operation is expressed by a set of methods involving the particles *event* and these methods are used to facilitate the evolution of the negotiation.

We propose the following methods associated to the particles *event* to model the cloning of an atom where new message particles are introduced:

- The method *Propose* is associated to the particle event *clone_propose(Id, New_id, Msg)* and models the introduction of a new proposal (*clone_propose*), made by one of the participants to the negotiation.

This method is expressed:

name(Id) @ enable @ clone_propose(Id, New_ id, Msg)<>- (enable @ name(Id)) & (freeze @ name(New_ id) @ propose(Rname, Content))

The atom identified by the particle *name(Id)* is cloned. The new proposal contained in the particle *propose(Rname, Content)* will be introduced in the new atom *name(New_id)*.

- The method *Accept* is associated to the event particle *clone_accept(Id, New_id, Msg)* and models the case when one of the participants has sent a message of acceptance of an older proposal (*clone_accept*).

This method is expressed:

name(Id) @ enable @ clone_accept(Id, New_ Id, Msg) <>- (enable @ name(Id)) & (freeze @ name(New_ Id) @ accept(Rname))

The atom identified by the *name(Id)* is cloned. The acceptance message contained in the particle *accept(Rname)* will be introduced in the new atom *name (New_id)*.

- The method *Reject* is associated to the event particle *clone_reject(Id, New_id, Msg)* and models the denial of an older proposal (*clone_reject*) made by one of the participants.

This method is expressed:

name(Id) @ enable @ clone_reject(Id, New_ Id, Msg) <>- (enable @ name(Id)) & (freeze @ name(New_ Id) @ reject(Rname))

The atom identified by the particle *name(Id)* is cloned. The refusal message contained in the particle *reject(Rname)* will be introduced in the new atom *name(New_ id).*

- The method *Create* is associated to the event particle *clone_create(Id, New_id, Msg).* This method models the invitation of a new sequence (*clone_create*) made by one of the participants for sharing the newly created negotiation phase.

This method is expressed:

name(Id) @ enable @ clone_create(Id, New_Id, Msg) @ <>- (enable @name(Id)) & (freeze @ name(New_Id) @ create(Rname, Type))

The atom identified by the particle *name(Id)* is cloned, and a particle *create(Rname, Type)* is introduced in the new atom *name(New_ id)* that will further generate the occurrence of a new representation particle for the new sequence participating in the negotiation.

These methods are described in a generic way. Thus, new particles may be added depending on how the current sequence builds negotiation graphs.

By these methods of the event particles, the duplication of an atom has been modeled, in which new message particles are introduced. In the new atom, the representation particles for the current negotiation phase remain identical with those of the first atom.

4. The Collaborative Negotiation Architecture

The main objective of this software infrastructure is to support collaborating activities in virtual enterprises. In VE partners are autonomous companies with the same object of activity, geographically distributed.

Taking into consideration, the constraints imposed by the autonomy of participants within VE, the only way to share information and resources is the negotiation process.

Figure 3 shows the architecture of the collaborative system:

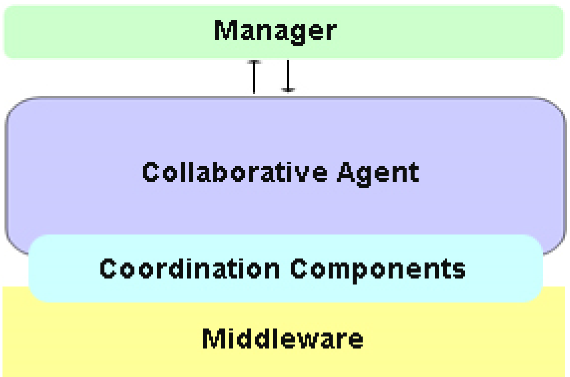


Fig. 3. The architecture of the collaborative system

This infrastructure is structured in ***four*** main ***layers***[1]: Manager, Collaborative Agent, Coordination Components and Middleware. A first layer is dedicated to the Manager of each organization of the alliance. A second layer is dedicated to the Collaborative Agent who assists its gas station manager at a global level (negotiations with different participants on different jobs) and at a specific level (negotiation on the same job with different participants) by coordinating itself with the Collaborative Agents of the other partners through the fourth layer, Middleware[2]. The third layer, Coordination Components, manages the coordination constraints among different negotiations which take place ***simultaneously***.

A Collaborative Agent aims at managing the negotiations in which its own gas station is involved (e.g. as initiator or participant) with different partners of the alliance.

Each negotiation is organized in three main steps: initialization; refinement of the job under negotiation and closing[3]. The initialization step allows to define what has to be negotiated (Negotiation Object) and how (Negotiation Framework)[4]. A selection of negotiation participants can be made using history on passed negotiation, available locally or provided by the

[1] Cretan A., Coutinho C., Bratu B. and Jardim-Goncalves R., (2011), *A Framework for Sustainable Interoperability of Negotiation Processes.* Paper submitted to INCOM'12 14th IFAC Symposium on Information Control Problems in Manufacturing.

[2] Bamford J.D., Gomes-Casseres B., and Robinson M.S., (2003), *Mastering Alliance Strategy: A Comprehensive Guide to Design, Management and Organization*. San Francisco: Jossey-Bass, pp. 27-38

[3] Sycara K., (1991), *Problem restructuring in negotiation*, in Management Science, 37(10), pp.24-32.

[4] Smith R., and Davis R., (1981), *Framework for cooperation in distributed problem solving*. IEEE Transactions on Systems, Man and Cybernetics, SMC-11, pp. 42-57.

negotiation infrastructure[5]. In the refinement step, participants exchange proposals on the negotiation object trying to satisfy their constraints[6]. The manager may participate in the definition and evolution of negotiation frameworks and objects[7]. Decisions are taken by the manager, assisted by his Collaborative Agent[8]. For each negotiation, a Collaborative Agent manages one or more negotiation objects, one framework and the negotiation status. A manager can specify some global parameters: duration; maximum number of messages to be exchanged; maximum number of candidates to be considered in the negotiation and involved in the contract; tactics; protocols for the Collaborative Agent interactions with the manager and with the other Collaborative Agents [9].

5. Coordination Components

In order to handle the complex types of negotiation scenarios, we propose five different components[10]:

- *Subcontracting* (resp. *Contracting*) for subcontracting jobs by exchanging proposals among participants known from the beginning;
- *Block* component for assuring that a task is entirely subcontracted by the single partner;
- *Divide* component manages the propagation of constraints among several slots, negotiated in parallel and issued from the split of a single job;
- *Broker*: a component automating the process of selection of possible partners to start the negotiation;
- *Transport* component implements a coordination mechanism between two ongoing negotiations in order to find and synchronize on the common transport of both tasks.

These components are able to evaluate the received proposals and, further, if these are valid, the components will be able to reply with new proposals constructed based on their particular coordination constraints[11].

From our point of view the coordination problems managing the constraints between several negotiations can be divided into two distinct classes of components:

- Coordination components in closed environment**:** components that build their images on the negotiation in progress and manage the coordination constraints according to information extracted only from their current negotiation graph (*Subcontracting, Contracting, Block, Divide)*;
- Coordination components in opened environment: components that also build their images on the negotiation in progress but they manage the coordination constraints according to available information in data structures representing certain characteristics of other negotiations currently ongoing into the system (*Broker, Transport).*

Following the descriptions of these components we can state that unlike the components in closed environment (*Subcontracting, Contracting, Block, Divide)* that manage the coordination constraints of a single negotiation at a time, the components in opened environment (*Broker, Transport)* allow the coordination of constraints among several different negotiations in parallel[12].

The novelty degree of this software architecture resides in the fact that it is structured on four levels, each level approaching a particular aspect of the negotiation process. Thus, as opposed to classical architectures which achieve only a limited coordination of proposal exchanges which take place during the same negotiation, the proposed architecture allows approaching complex cases of negotiation coordination. This aspect has been accomplished through the introduction of coordination components level, which allows administrating all simultaneous negotiations in which an alliance partner can be involved.

The coordination components have two main functions such as: i) they mediate the transition between the negotiation image at the Collaboration Agent level and the image at the Middleware level; ii) they allow implementing various types of appropriate behavior in particular cases of negotiation. Thus we can say that each component corresponding to a particular negotiation type.

Following the descriptions of this infrastructure we can state that we developed a framework to describe a negotiation among the participants to a virtual enterprise. To achieve a generic coordination framework, nonselective and flexible, we found necessary to first develop the structure of the negotiation process that helps us to describe the negotiation in order to establish the general environment where the participants may negotiate. To develop this structure, we proposed a succession of phases that are specific to different stages of negotiation (initialization, negotiation, contract

---

[5] Zhang X. and Lesser V., (2002), *Multi-linked negotiation in multi-agent systems*. In Proc. of AAMAS, Bologna, pp. 1207 – 1214.

[6] Barbuceanu M. and Wai-Kau Lo, (2003), *Multi-attribute Utility Theoretic Negotiation for Electronic Commerce*. In AMEC III, LNAI, pp. 15-30.

[7] Keeny R. and Raiffa H., (1976), *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. JohnWilley & Sons.

[8] Bui V. and Kowalczyk R., *On constraint-based reasoning in e-negotiation agents*. In AMEC III, LNAI 2003, pp. 31-46.

[9] Faratin P., (2000), *Automated service negotiation between autonomous computational agent*. Ph.D. Thesis, Department of Electronic Engineering Queen Mary & West-field College.

[10] Cretan A., Coutinho C., Bratu B. and Jardim-Goncalves R., (2011), *A Framework for Sustainable Interoperability of Negotiation Processes.* Paper submitted to INCOM'12 14th IFAC Symposium on Information Control Problems in Manufacturing.

[11] Vercouter, L., (2000), *A distributed approach to design open multi-agent system*. In 2nd Int. Workshop Engineering Societies in the Agents' World (ESAW), pp. 32-49.

[12] Muller H., (1996), *Negotiation principles.* Foundations of Distributed Artificial Intelligence.

adoption) that provided a formal description of the negotiation process.

The advantage of this structure of the negotiation process consists on the fact that it allows a proper identification of the elements that constitute the object of coordination, of the dependencies that are possible among the existing negotiations within the VE, as well as the modality to manage these negotiations at the level of the coordination components.

The negotiation process involves several parties (for several bilateral negotiations), each having different criteria, constraints and preferences that determine their individual areas of interest[13]. Criteria, constraints and preferences of a participant are partially or totally unknown to the other participants. The job under negotiation is described as a multi-attribute object. Each attribute is related to local constraints and evaluation criteria, but also to global constraints drawing dependencies with other attributes[14].

In conclusion, the proposed architecture provides the following features:

- to define the negotiation process structure: participants, interaction protocol, negotiation protocol, tactics and coordination components, the negotiation object and the negotiation strategies;
- the modeling all negotiations for a gas station in the form of a set of bilateral negotiations, which the agent can operate independently;
- the modeling of the coordination among the negotiations based on a set of coordination components and the synchronization mechanisms at the middleware level.

Thus, we can say, that we have proposed an infrastructure that manages, in a decentralized manner, the coordination of multi-phase negotiations on a multi-attribute object and among a lot of participants.

Conclusions

This paper proposes an intelligent mechanism for modeling and managing parallel and concurrent negotiations. The business-to-business interaction context in which our negotiations take place forces us to model the unexpected and the dynamic aspects of this environment. An organization may participate in several parallel negotiations. Each negotiation may end with the acceptance of a contract that will automatically reduce the available resources and it will modify the context for the remaining negotiations. We have modeled this dynamic evolution of the context using IAMs metaphor that allows us to limit the acceptance of a negotiation to the available set of resources. The proposed negotiation infrastructure aims to help the different SMEs to fulfil their entire objectives by mediating the collaboration among the several organizations gathered into a virtual enterprise.

A specific feature that distinguishes the negotiation structure proposed in this work from the negotiations with imposed options (acceptance or denial) is that it allows the modification of the proposals through the addition of new information (new attributes) or through the modification of the initial values of certain attributes (for example, in the case of gas stations the gasoline price may be changed).

In the current work we have described in our collaborative mechanism only the interactions with the goal to subcontract or contract a task. A negotiation process may end with a contract and in that case the supply schedule management and the well going of the contracted task are both parts of the outsourcing process.

In order to illustrate our approach we have used a sample scenario where distributed gas stations have been united into virtual enterprise. ***Take into consideration this*** scenario**,** one of the principal objectives was related to the generic case and means that this proposed infrastructure can be used in other activity domains.

Regarding research perspective continuation, we will focus on the negotiation process and the coordination process taking into consideration the contracts management process. In this way the coordination can administrate not only the dependence between the negotiations and the contracts which are formed and with execution dependences of those contracts.

References

- Camarinha-Matos L.M. and Afsarmanesh H., *Collaborative Networked Organizations*, 2004 Kluwer Academic Publisher Boston
- Singh M.P., *Commitments among autonomous agents in information-rich environments*. In Proceedings of the 8th European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW), pp. 141–155, May 1997
- Bamford J.D., Gomes-Casseres B., and Robinson M.S., Mastering Alliance Strategy: A Comprehensive Guide to Design, Management and Organization. San Francisco: Jossey-Bass, 2003
- Sycara K., *Problem restructuring in negotiation*, in Management Science, 37(10), 1991
- Smith R., and Davis R., *Framework for cooperation in distributed problem solving*. IEEE Transactions on Systems, Man and Cybernetics, SMC-11, 1981

[13] Schumacher M., (2001), *Objective coordination in multi-agent system engineering – design and implementation.* In Lecture Note in Artificial Intelligence, No. 2093, Springer Verlag, pp.72-88.

[14] Kraus S., (2001), *Strategic negotiation in multi-agent environments.* MIT Press, pp.56-67.

- Zhang X. and Lesser V., *Multi-linked negotiation in multi-agent systems*. In Proc. of AAMAS 2002 July, Bologna, pg. 1207 – 1214
- Barbuceanu M. and Wai-Kau Lo, Multi-attribute Utility Theoretic Negotiation for Electronic Commerce. In AMEC III, LNAI 2003, pg. 15-30
- Keeny R. and Raiffa H., Decisions with Multiple Objectives: Preferences and Value Tradeoffs. JohnWilley & Sons, 1976
- Bui V. and Kowalczyk R., *On constraint-based reasoning in e-negotiation agents*. In AMEC III, LNAI 2003, pp. 31-46
- Faratin P., *Automated service negotiation between autonomous computational agent*. Ph.D. Thesis, Department of Electronic Engineering Queen Mary & West-field College, 2000
- Cretan A., Coutinho C., Bratu B. and Jardim-Goncalves R., *A Framework for Sustainable Interoperability of Negotiation Processes.* Paper submitted to INCOM'12 14th IFAC Symposium on Information Control Problems in Manufacturing, 2011
- Cretan, A., Coutinho, C., Bratu, B., and Jardim-Goncalves, R., *NEGOSEIO: A Framework for Negotiations toward Sustainable Enterprise Interoperability.* Annual Reviews in Control, 36(2): 291–299, Elsevier, ISSN 1367-5788, 2012. http://dx.doi.org/10.1016/j.arcontrol.2012.09.010
- Vercouter, L., *A distributed approach to design open multi-agent system*. In 2nd Int. Workshop Engineering Societies in the Agents' World (ESAW), 2000
- Muller H., *Negotiation principles.* Foundations of Distributed Artificial Intelligence, 1996
- Hurwitz, S.M., Interoperable Infrastructures for Distributed Electronic Commerce. 1998
- Hu, J., Deng, L., "An Association Rule-Based Bilateral Multi-Issue Negotiation Model," in *Computational Intelligence and Design (ISCID),* Fourth International Symposium, Volume 2, pp. 234 – 237, 2011
- Luo, X., Miao, C., Jennings, N., He, M., Shen, Z. and Zhang, M.," KEMNAD: A Knowledge Engineering Methodology for Negotiating Agent Development," *Computational Intelligence* ., 28 (1), 51-105, 2012
- Robinson W., and Volkov V., *Supporting the negotiation life cycle.* Communications of the ACM, 1998
- Ossowski S., *Coordination in Artificial Agent Societies*. Social Structure and its Implications for Autonomus Problem-Solving Agents, No. 1202, LNAI, Springer Verlag, 1999
- Schumacher M., *Objective coordination in multi-agent system engineering – design and implementation.* In Lecture Note in Artificial Intelligence, No. 2093, Springer Verlag, 2001
- Sycara, K., Dai, T.,"Agent Reasoning in Negotiation," *Advances in Group Decision and Negotiation*, Volume 4, Handbook of Group Decision and Negotiation, Part 4, pp. 437-451, 2010
- Kraus S., Strategic negotiation in multi-agent environments. MIT Press, 2001