

Michał Piotrowski, Henryk Krawczyk

Gdansk University of Technology

Faculty of Electronics, Telecommunications and Informatics, POLAND

bastian@eti.pg.gda.pl, hkrawk@eti.pg.gda.pl

One of the most important aspects of collaboration between people or organizations is effective communication. To support different kinds of human activities (e. g. negotiations), general interaction procedures need to be defined. Scenarios of such activities can be expressed in choreography languages. In the paper WS-CDL language is used to model buy/sell negotiations. The suitable method for generation of scenarios is given and a concrete scenario is analyzed. Moreover, testbed for empirical experiments is presented and some experiment results are discussed.

1. INTRODUCTION

Negotiation is a human activity that occurs between at least two parties (negotiators) and concerns a concrete subject such as selling/buying products, choosing the best solution to a given problem or finding the most suitable services, etc. In the case of distributed computing environments, agents as well as people can be negotiators. Agents are special autonomous and mobile programs which seek either other agents or people to negotiate with them on behalf of their owners. The latter kind of negotiations is called electronic negotiations or briefly e-negotiations.

Let $h2h$ denote human negotiations. Human negotiations can be performed in real and virtual environments. Furthermore, let $e2e$ denote electronic negotiations, where $e2e = a2a$ (a denotes agent or algorithm) or $e2e = a2h$ which denotes humans negotiating with some kind of computer program.

Negotiations can be described by a computational negotiation model. The model proposed in (Krawczyk-Brylka, 2008) is suitable for all types of negotiations. It consists of initial conditions of negotiations, negotiation strategies and negotiation outcomes. Initial conditions describe the subject of negotiation, e. g. purchase of medical equipment, negotiation attributes, e. g. price, delivery date, payment date etc. and an area of expectable negotiation outcomes, e. g. price between minimal and maximal possible values. The model describes the negotiation process as a series of requests (demands) which form the negotiation dance. The demands represent values of negotiation attributes proposed by negotiation partners. The graphical representation of such a sequence is a polygonal curve.

The paper concentrates on $a2h$ negotiations considering experiments in which purchasing of medical equipment was negotiated. First of all we try to design an electronic negotiator which can behave according to the given negotiation scenario.

The paper describes the behaviour of the negotiator on two levels. The low level behaviour is described by negotiation dance as sequences of values of negotiation attributes. The high level behaviour corresponds to scenarios described by both modeling and choreography languages, which present sequences of demands/requests messages. In consequence, our e-negotiator is an algorithm which behaves appropriately to the given negotiation task, similarly to a human negotiator. Finally, the paper presents the main principles of the algorithm behaviour, the testbed for a2h negotiation experiments and conclusions.

2. LOW LEVEL DESCRIPTION OF NEGOTIATION SCENARIOS

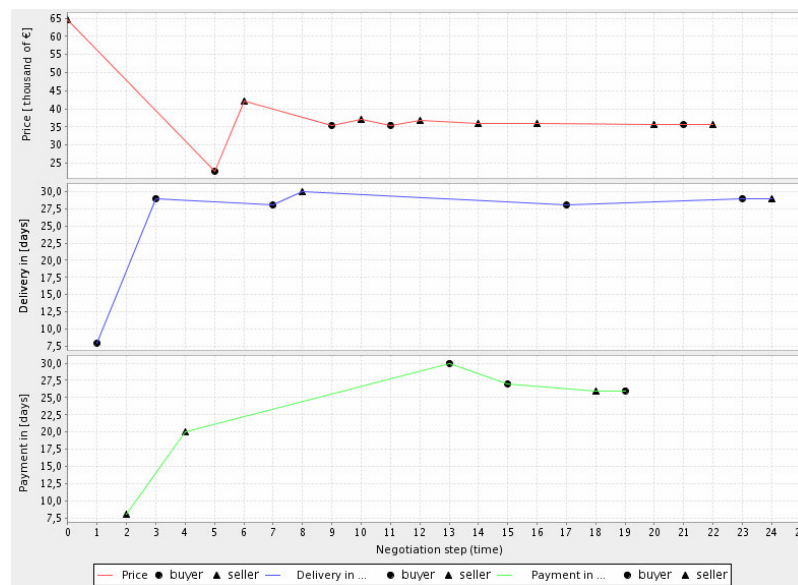


Figure 1 - A sample of h2h negotiation dance

Low level description of negotiation scenarios is related to negotiation dances. Figure 1 presents a negotiation dance obtained in a negotiation experiment. The negotiation dance presents changes of demands in time. In figure 1 one can observe changes in three negotiated attributes. Buyer demands are marked with circle and seller demands are marked with triangle. Price, delivery date and payment date were the attributes considered by negotiators participating in the negotiation experiments. By analyzing negotiation dances it is possible to determine relationships between negotiators' demands.

The proposed negotiating algorithm is based on the model in (Krawczyk B., 2008) and uses parameters obtained from both analyses of the negotiation dances and the obtained negotiation experiments outcomes. The algorithm is constructed to generate suitable requests and responses to a partner in a negotiation. The algorithm

takes into account the history of the negotiation in the form of a negotiation dance. The principles of the algorithm are as follows:

1. Its first demand (request) is generated as the best value of a negotiation attribute from its point of view.
2. Its subsequent demands (responses), depend on the history of partner's propositions, specifically the algorithm makes concessions to a similar degree as its partner.

In the process of demands generation, proposed values of attributes are modified by a Gauss distribution with μ equal to the generated attribute value and σ calculated on the basis of statistical data obtained in h2h negotiation experiments. The algorithm is tuned to maximize effectiveness while using cooperative style of negotiations (Fisher, 1991).

The proposed negotiating algorithm is used in a virtual environment as one of the partners in a2h negotiations.

3. HIGH LEVEL DESCRIPTION OF NEGOTIATION SCENARIOS

The negotiating algorithm itself is not sufficient to perform a2h negotiations. Communication procedures and different behaviour patterns should be defined. High level description of negotiation scenarios specifies communication patterns from the global point of view. Choreography languages can be used to describe such interaction scenarios. In general, such languages concentrate on visible aspects of negotiation only, ignoring the logic of parties involved (Peltz, 2003). Recently, languages based on web services architecture have become very popular with the emergence of service oriented architecture (SOA). SOA facilitates integration of different business systems and desktop applications. One of the most important choreography languages associated with SOA is Web Services Choreography Description Language (WS-CDL) (Kavantzias, 2005).

Based on h2h negotiation experiments, a general negotiation scenario has been modeled as a UML sequence diagram (figure 2). In the negotiation experiments two roles were defined: buyer and seller. The goal of the negotiations was to obtain a contract for purchase of medical equipment. The actors in figure 2, buyer (B) and seller (S) negotiate until both of them accept each other's terms (loop box) or when one of the negotiators breaks the negotiation (treated as an exception). The negotiation process consists of exchanges of demands. In each step of the negotiation, one party presents their demands to the other. The order of demands is arbitrary, e. g. more than one set of demands from one of the participants can come in succession or the participants present their demands alternately (alt box). The demands are presented as asynchronous method calls in UML notation, since WS-CDL is based on the concept of web services. "Demands", the argument of *propose()* method, represents a set of values of negotiation attributes, e. g. *Demands* = {price = 50000 €, delivery in 14 days}.

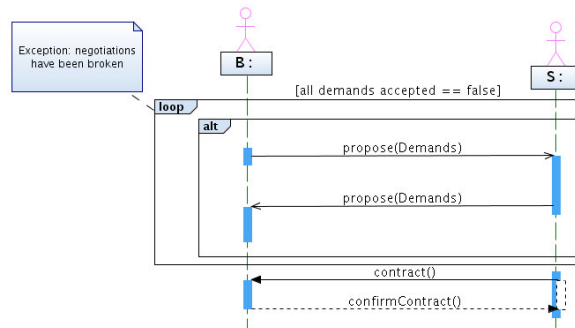


Figure 2 – General high-level scenario of a buy/sell negotiation represented as UML sequence diagram

The next step in the choreography modeling process is the formal notation of the interactions in WS-CDL language. Using a WS-CDL document, the system can manage the negotiation process. Listing 1 presents a section of a WS-CDL document describing interactions modeled in figure 2.

Before interactions can be expressed in WS-CDL language, information types should be defined. Information types represent kinds of information or documents exchanged in all communication processes between parties involved in a negotiation. Parameter “*Demands*” has to be described as an XML Schema type. Afterwards variables containing the exchanged tokens of information have to be defined.

In listing 1 workunit named *PresentingDemands* corresponds to the loop box in the UML diagram, choice corresponds to the alt box in the diagram and interaction named *demandsFromSeller* corresponds to the situation where a seller presents his demands to a buyer.

Listing 1 – Section of WS-CDL document representing modeled scenario

```

<choreography name="Negotiations" root="true">
  ...
  <workunit name="PresentingDemands"
    repeat="not (cdl:getVariable('BuyerAgrees'),'','') or
      not (cdl:getVariable('SellerAgrees'),'','') ">
    <choice>
      <interaction name="demandsFromSeller"
        channelVariable="tns:SellerToBuyer"
        operation="propose">
        <participate fromRoleTypeRef="tns:SellerRole"
          toRoleTypeRef="tns:BuyerRole"
          relationshipType="tns:SellerAndBuyer" />
        <exchange action="request" name="sendDemandsToBuyer"
          informationType="tns:demndsList">
          <send
            variable="cdl:getVariable('SellerDemands','','') " />
          <receive
            variable="cdl:getVariable('SellerDemands','','') " />
        </exchange>
        <exchange action="respond" name="areDemandsAccepted"
          informationType="tns:boolean">
          <send

```

```

        variable="cdl:getVariable('BuyerAgrees','','') " />
    <receive
        variable="cdl:getVariable('BuyerAgrees','','') " />
    </exchange>
</interaction>
<interaction name="demandsFromBuyer"
    channelVariable="tns:BuyerToSeller"
    operation="propose">
    ...
</interaction>
</choice>
</workunit>
<workunit name="Contract" ...> ... </workunit>
<exceptionBlock name="HandleExceptions"> ... </exceptionBlock>
</choreography>

```

4. TESTBED FOR A2H NEGOTIATION EXPERIMENTS

The next step is to deploy the choreography in an appropriate environment. Figure 3 presents such an environment. It consists of two main parts: GAJA system and e-negotiator module. The e-negotiator module integrates a WS-CDL scenario with the algorithm simulating a negotiator. A WS-CDL choreography is used to manage the course of negotiations.

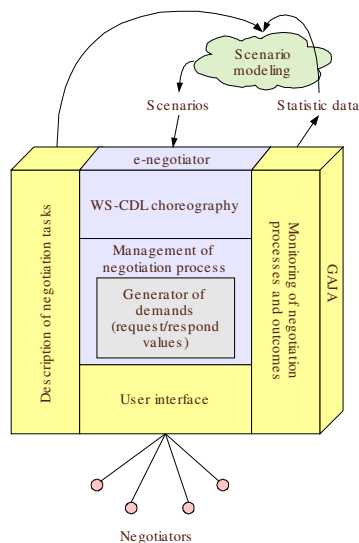


Figure 3 - Architecture of negotiation scenario testbed

The GAJA system provides descriptions of negotiation tasks for the negotiation algorithm and human negotiators. The negotiation algorithm obtains negotiation attributes and initial conditions from the GAJA system as appropriate algorithm parameters. Human negotiator is presented with a precise description of negotiation tasks, including his role in the negotiation experiment, negotiation goals etc.

Moreover, GAJA provides a user interface which acts as a middleware between the e-negotiator and a human negotiator. Figure 4 presents a screenshot of the user interface provided by the GAJA system. In the experiments, the following task was defined: a pair of negotiators had to negotiate the contract for purchase of medical equipment. The contract consisted of the price, delivery and payment dates.

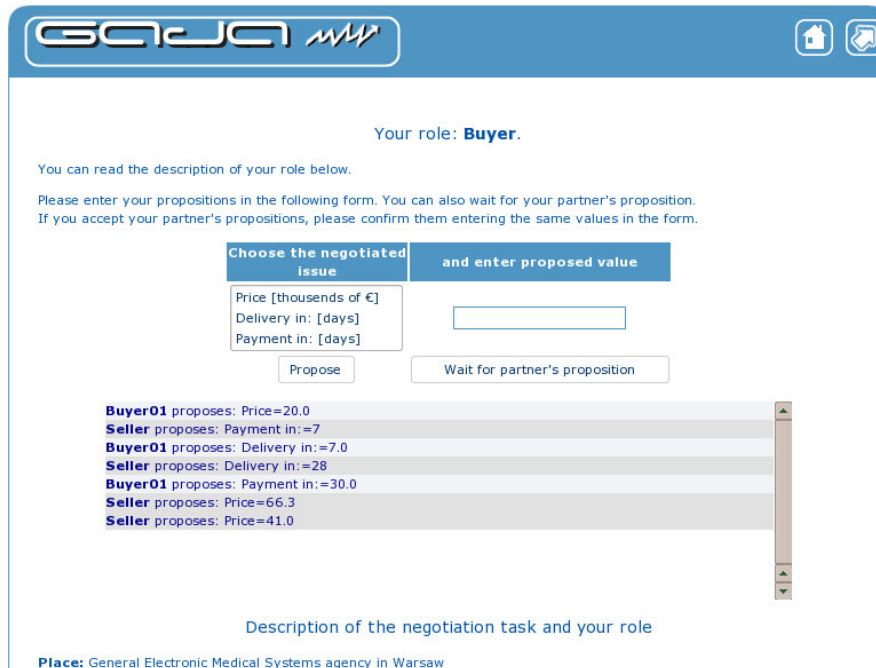


Figure 4 – Screenshot of user interface for a2h negotiations

Parameters observable in negotiation dance are used in the quality evaluation of negotiation experiments. To estimate the quality of the e-negotiator, the modified model proposed in (Krawczyk-Brylka, 2008) is used. The main quality attributes considered for comparing h2h and a2h negotiations are as follows:

- effectiveness – percent of negotiations which ended with a contract which satisfied both negotiators;
- efficiency – based on the time of negotiation measured by the number of steps of the negotiation (number of interactions between negotiators). The lower the number of interactions, the higher the efficiency;
- main negotiation style – negotiation style can be competitive, collaborative or balanced (Fisher, 1991). The collaborative style is assumed to be preferred. The evaluation of the negotiation style is based on the degree of concessions made by negotiators observed in the negotiation dance. In the negotiation dance graph it is possible to observe the change of demands values in each step of the negotiation and compare it to concessions made by a partner in former steps of the negotiation.

Table 1 – Comparison of h2h and a2h negotiation quality

<i>Quality attribute</i>	<i>h2h negotiations</i>	<i>a2h negotiations</i>
effectiveness	broken negotiations - 6.25%	broken negotiations - 27.5%
efficiency	avg. 41 negotiation steps	avg. 28 negotiation steps
negotiation style	73.14% of negotiators used the collaborative style	- 72% of human negotiators used the collaborative style - the computer algorithm always used the collaborative style

50 h2h and 50 a2h experiments were conducted. The participants of a2h experiments had participated in h2h negotiation experiments as well, therefore the results of a2h experiments are directly comparable to the results of h2h negotiations. H2h and a2h negotiation experiments were performed using the GAJA system. In the case of h2h negotiations the communication medium was a chat.

Table 1 summarizes the quality comparison of h2h and a2h negotiations. The results of the experiments were as follows: the efficiency of a2h negotiations is 1/3 higher than h2h negotiations due to the very formal style of communication imposed by the choreography and the user interface. However, the number of broken a2h negotiations was 4 times higher comparing to experiments where both of the negotiators were human. The algorithm assumed that the typical partner will use the cooperative style of negotiations and broke the negotiations if the partner's style deviated from the preferred style of negotiations to a high degree. The preferred negotiation style in both negotiation types was the collaborative style.

A survey was conducted after the a2h negotiation experiments. The main conclusion of the survey was that human negotiators need a broader context of the partner's demands. Human negotiators tend to test the limits of concessions that their partner is willing to make. The algorithm does not warn when it decides that partner's demands are unreasonable and breaks the negotiations immediately. In consequence, human negotiators who wanted to test their partners' limits, accidentally caused the algorithm to make the decision to break the negotiation.

Therefore, interactions representing context information are needed in a2h negotiations. This context information can be a warning for a partner that his demands are unreasonable or that a negotiator is satisfied with the course of negotiations. The algorithm can easily be updated to present this kind of information based on its state and a dictionary of contextual phrases. This improvement can be modeled as a WS-CDL document and integrated into testbed.

5. CONCLUSIONS

Modelling negotiations on a low level is not sufficient. The context of negotiations is a very important factor for successful negotiations. The context can be understood as a business context of the negotiation environment. This knowledge is the basis for

proper decisions. Additionally, the context can be understood as a conversational context needed by people to properly understand their partner (Strecker, 2006).

Conversational context can be modeled as WS-CDL documents, which facilitates integration of improvements in e-negotiation systems. WS-CDL can be translated to WS-BPEL notation or a SOA module which can be directly integrated and executed in one of the available SOA middlewares. The initial tests for these procedures were conducted and proved successful (Piotrowski, 2008).

It is possible to automatically translate UML modeled diagrams into executable XML-based languages like WS-BPEL or WS-CDL (Benyoucef, 2005). The next step of development would be to use dedicated modelling environments for orchestration and choreography languages and omit the UML modelling stage. Graphical modelling tools for WS-CDL are in development (e. g. Pi4SOA project). WS-CDL can be used to model negotiations from the global point of view, so it is suitable for integrating models focused on local (negotiator's) point of view. Moreover WS-CDL standard is an information driven language and as such, it facilitates integration of advanced protocols, e. g. states in a social protocol (Picard, 2006) can be directly modelled as WS-CDL variables and any associated logic for such a protocol can be added as a set of web services deployed in a middleware for WS-CDL execution.

In this manner it would be possible to directly modify scenarios in WS-CDL language, deploy a new version of the scenario in a middleware and immediately test the results of changes.

5.1 Acknowledgments

This work was supported under Ministry of Science and Higher Education research project no N516 035 31/3499.

6. REFERENCES

1. Benyoucef M., Rinderle S. A Model-driven Approach for the Rapid Development of E-negotiation Systems. School of Management, University of Ottawa École de gestion, Université d'Ottawa, 2005.
2. Fisher R. and Ury W. and Patton, B. Getting to Yes: Negotiating Agreement Without Giving in. Houghton Mifflin Books, 1991
3. Kavantzaz N., Burdett D., Ritzinger G., Fletcher T., Lafon Y., Barreto C., WSCDL: Web Service Choreography Description Language, www.w3c.org, November 2005
4. Krawczyk-Brylka B. Piotrowski M. "Using a computational model to compare objective negotiations in real and virtual environments". In International Journal of Production Research, vol 46; number 5, Taylor & Francis 2008: 1315-1333
5. Peltz C. "Web services orchestration and choreography". In Computer vol. 36, number 10, 2003: 46-52
6. Picard W. "Computer Support for Adaptive Human Collaboration with Negotiable Social Protocols", Proceedings of the 9th int. Conference on Business Information Systems, Klagenfurt, Springer Boston 2006, 353-360
7. Pi4SOA Project homepage, <http://www.pi4tech.com>
8. Piotrowski M. and Krawczyk-Brylka B. "The Web Based System for Recording and Analysing Different Kinds of Negotiations". In WEBIST 2006, Proceedings of the Second International Conference on Web Information Systems and Technologies: Society, e-Business and e-Government / e-Learning, Setubal, INSTICC Press, Portugal, 2006: 82-87
9. Piotrowski M., "Creating negotiation scenarios using WS-CDL language", to be published in Polish in KASKBook, Gdansk 2008
10. Strecker S., Kersten G., Kim J., Law K. "Electronic Negotiation Systems: The Invite Prototype". In Proceedings of the Collaborative business MKWT'06, Germany 2006