Annals of the CIRP, Vol.53/1, pp.122-125, 2004

Agent Based Negotiation for Collaborative Design Decision Making

Y. Jin[1], S. C-Y. Lu[1] (1)
[1]IMPACT Laboratory, Dept. of Aerospace and Mechanical Engineering
University of Southern California, Los Angeles, USA

Abstract
Complex design involves trade-offs and teamwork. To make collaborative design decisions, designers must negotiate with each other to resolve their discrepancies through exploring the design space, generating new ideas and compromising for agreement. Advances in negotiation research have been made in social psychology, distributed artificial intelligence, and decision theory. But few have been applied to design. Built on existing negotiation research, we are developing an agent based negotiation framework to support collaborative design decision-making. This paper presents the negotiation protocol, information models, and negotiation strategies of the framework followed by discussion of an application case example.

Keywords:
Design, Decision-making, Agent

1 INTRODUCTION

Engineering decision-making is about trade-offs among various competing design criteria. In automotive body structure design, for example, an engineer must, when making design decisions, balance between vehicle performance, safety, and manufacturing cost. On the other hand, engineering decision-making is about teamwork. It always involves multiple engineers and engineering systems specializing in different functional disciplines. Engineers work together not only to define problems and identify requirements, but also negotiate with each other to balance their local decisions, generate ideas, and develop globally acceptable solutions.

Most existing collaborative engineering support systems aim at providing seamless information flows among engineering systems. Shared database systems, various communication protocols, and workflow systems have been developed to facilitate information sharing, design change propagation, and process management. Few systems provide means for engineers to negotiate their decisions for the interest of overall design. In our research, we take an agent-based approach to support collaborative design decision-making. The goal is to develop an agent-based negotiation framework that links designers and engineering systems together at decision-level and facilitate negotiation among them.

Engineering as collaborative negotiation (ECN) has recently attracted attentions from both academic and industrial researchers [1]. Negotiation in general is a process in which a joint decision is made by two or more parties [2]. The parties first verbalize contradictory demands and then move towards agreement by a process of concession-making or search for new alternatives. For multi-criteria collaborative design problems, negotiation is a way for multiple designers to exchange information, acquire knowledge of other designers' perspectives and intents, and identify new opportunities based on the learned information and knowledge. Therefore, design negotiation is not only a way to reach mutually acceptable solutions, but also a method to create new opportunities and new designs.

In this paper, we present an agent-based negotiation framework (ANF) for collaborative design decision-making. Built on previous research on negotiation and multi-agent systems, ANF is composed of a negotiation protocol, a set of negotiation strategies, and a network of intelligent agents that help human designers and computer systems to follow the negotiation protocol, select negotiation strategies, and make proposals. In the following, we first review the research on negotiation and discuss the requirements of engineering design negotiation. After that we present our framework and discuss a case example. Finally, we draw concluding remarks in the last section.

2 NEGOTIATION RESEARCH AND REQUIREMENTS OF DESIGN NEGOTIATION

Negotiation has been a research topic in various areas including social psychology where the focus is on developing principles for successful human interactions [2][3][4], distributed artificial intelligence (DAI) of which the goal is to develop protocols and reasoning mechanisms for computers to work together [5][6], and decision-theory that promotes decision-theoretic and game-theoretic normative models of negotiation [7].

Gulliver [3] proposed an eight-phase model of negotiation process that describes the progress of negotiation from the initial recognition of the dispute to some kind of outcome. The eight phases are: search for arena, agenda setting, exploring the filed, narrowing the difference, preliminaries to final bargaining, final bargaining, ritual affirmation, and execution. Pruitt [2] proposed a strategic choice model of negotiation, stating that parties involved in negotiation must make strategic choices at every point in time. The choices include conceding unilaterally, standing firm, or collaborating with other parties in search of a mutually acceptable solution. Toulmin [4] suggested a simple model of argument structure for negotiation. He proposed that the first step in an argument is for one party to express an opinion, called "claim". If the claim is challenged, it has to be defended by "data" and "warrant" successively. DAI researchers have developed various methods for distributed cooperative agents, or computer systems, to reconcile their disparities. Sycara [5] proposed a negotiation process that uses case-based reasoning mechanism together with a restricted protocol to support agents resolving their goal conflicts. Parsons et al [6] developed a negotiation protocol in which agents explore mutual spaces of negotiation and eventual arrive at a mutually acceptable solution by continuously exchanging arguments. Extending the multi-objective decision theory and game theory, Raiffa [7] examined the dynamics of win-lose, win-win and multi-party negotiations and proposed novel approaches for successful negotiation.

While the advances of the above-mentioned negotiation research have been applied in business management activities and networked computer systems, few have been introduced to the field of engineering design. One main reason for this lack of progress may be due to the complexity of negotiation in engineering design. In order to support engineering design negotiation, one must address the following important requirements.

1. Deal with multi-functional disciplines. Designers may come from different functional areas and they may not understand each other well.
2. Deal with semi-structured and ill-structured problems. Design problems are often open-ended and not well represented. Human involvement is inevitable.
3. Allow a variety of engineering issues for negotiation. Design negotiation is not merely about conflict of parameter values. The issues must include function requirements, and design goals and preferences.
4. Encourage generating new alternatives. Unlike in many other fields where give-and-take is the key, design negotiation must encourage generating new ideas.
5. Deal with multiple interrelated negotiations. Engineering design often involves multiple interrelated negotiations between different groups of designers.

3 AGENT-BASED NEGOTIATION NETWORK

3.1 A General Negotiation Model

Negotiation involves two or more parties to raise incompatible issues and find ways to reconcile the differences. We model a given engineering negotiation situation as Equation (1) shown below.

$$N = [\boldsymbol{D}, \boldsymbol{I}, \boldsymbol{P}; \boldsymbol{S}, \boldsymbol{T}] \quad (1)$$

Where:

$\boldsymbol{D} = \{d_1, d_2, \ldots d_n\}$: A set of participants, i.e., designers and/or computer systems.

$\boldsymbol{I} = \{i_1, i_2, \ldots i_m\}$: A set of all types of issues that are negotiable.

$\boldsymbol{P} = [\boldsymbol{r}, \boldsymbol{q}, \boldsymbol{a}]$: A protocol composed of communicative speech acts $\boldsymbol{r}=\{r_1, r_2, \ldots r_l\}$, negotiation states $\boldsymbol{q}=\{q_1, q_2, \ldots q_k\}$, and strategic actions $\boldsymbol{a}=\{a_1, a_2, \ldots, a_s\}$.

$\boldsymbol{S} = \{s_1, s_2, \ldots s_g\}$: Strategies for choosing strategic actions, composed of a set of strategic rules.

$\boldsymbol{T} = \{t_1, t_2, \ldots t_h\}$: Tactics for choosing proposed issues instances, composed of a set of tactic rules.

For a given collaborative design situation, participants, negotiable issues and negotiation protocol are common knowledge to everyone, while different designers may have their own strategies and tactics. The research question here is how can we model negotiation issues, devise protocols, and develop strategies and tactics so that the above requirements can be satisfied?

3.2 Multi-Level Issues and Arguments

In engineering design, negotiation can be between designers, a designer and a manager, a customer and a designer, or a customer and a manager. A computer system can also be a party in a negotiation. While identifying participants in engineering negotiation is relatively straightforward, modelling "what are the issues to be negotiated" and "what kind of arguments can be made during negotiation" have significant impact on negotiation performance. In our research, we developed a multi-level model to represent negotiation issues, as shown in Equation (2) and Table 1.

$$I = \{ \boldsymbol{P_v}, \boldsymbol{C_p}, \boldsymbol{D_p}, \boldsymbol{F_r}, \boldsymbol{C_c}, \boldsymbol{C_s}, \boldsymbol{O_c}\} \quad (2)$$

Where

$\boldsymbol{P_v}, \boldsymbol{D_p}, \boldsymbol{F_r}, \boldsymbol{C_c}, \boldsymbol{C_s}, \boldsymbol{O_c}$ are explained in Table 1.

Issues	Examples
$\boldsymbol{O_c}$: Company Objectives	Safety and Performance
$\boldsymbol{C_s}$: System Criteria	Low cost, safety
$\boldsymbol{C_c}$: Component Criteria	Weight, opening torque
$\boldsymbol{F_r}$: Functional Requirement	Support Door
$\boldsymbol{D_p}$: Design Parameter	Hg, Ds
$\boldsymbol{C_p}$: Parametric Constraints	Hg < 0.5 Ds (Door size)
$\boldsymbol{P_v}$: Parameter-Value	Hinge position Hg=20cm

Table 1: Multi-Level Negotiable Issues and Examples

In ANF, issues range different levels, from bottom parameter value level to top company objective level, as shown in Table1. A lower-level issue is strongly influenced or governed by its higher-level issues. A parameter's value must be within a given range due to one or more parameter constraints. The parameter constraints exist because certain parameters are chosen. The parameters are chosen to satisfy certain functional requirements, and so on. Finally, company objectives become the governing factor for negotiation.

A negotiation often starts when two designers find discrepancies among their lower level issues. To resolve the discrepancies, they may exchange more information at that level. If this is not enough, they may move the issue to a higher level by making the related higher level issues negotiation objects. This "multi-level" negotiation is important for engineering design because the consistency of higher-level issues implies better global design.

To ensure that negotiation is efficient and moving to a right direction, designers should do more than simply "agree" or "reject". They must provide "argument" for others to know "what do I want" and "why." Following Toulmin (1969), we model argument as a structure depicted in Figure 2.

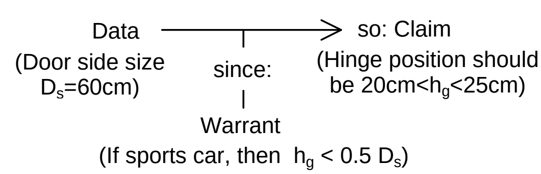


Figure 1: Toulmin's Argument Structure and Example

In this model, negotiation starts when a designer makes a "Claim." If the claim is challenged by another designer, then the designer adduces "Data" to defend it. If the challenger is not satisfied with the data, then a "Warrant" can be supplied by the designer, either voluntarily or at the request of the challenger. A "warrant" can be either a rule that states the relation between claim and data as shown in

Figure 1, or a related higher-level issue. In this case, if the challenger starts to challenge the “warrant”—i.e., the higher-level issue, the negotiation moves to a higher-level in which the “warrant” becomes a “Claim” and negotiation continues. Figure 2 is a BNF (Backus Naur Form) summary of our model of issues and arguments.

<Argument> ::= <Claim>* | <Claim> <Data> |
<Claim> <Data> <Warrant>

<Claim> ::= <Issue> IS <Value>|NOT<Issue>IS<Value>

<Data> ::= <Issue> HAS <Value>

<Warrant> ::= SINCE <Issue> | SINCE <Rule>

<Rule> ::= IF <Argument>|<Claim>|<Data>|<Warrant>
THEN < Argument>|<Claim>|<Data>|<Warrant>

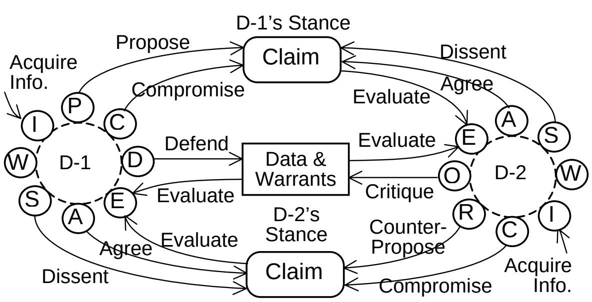
<Issue> ::= Issue belongs to I of Equation (2).

<Value> ::= Value assignable to corresponding issue

Figure 2: BNF of Argument Model of ANS

3.3 Speech-Act and Argumentation Based Negotiation Protocol

A negotiation protocol for engineering design should be flexible enough to deal with semi-structured or even ill-structured design problems in which not all issues and arguments can be numerically represented. On the other hand, it should also be “formal” enough so that negotiation efficiency can be achieved and computer support be attained. In our research, we developed a speech-act and argumentation based negotiation protocol based the above issue and argument model and a set of verbs chosen from speech-act dictionaries. Figure 3 depicts the protocol being applied in a two-party negotiation.



Speech-Acts: Propose, Agree, Dissent, Defend, Compromise, Critique, Counterpropose

Negotiation States: (P)=Proposing, (D)=Defending, (C)=Compromising, (A)=In-Agreement, (S)=In-Disagreement, (Q)=Critiquing, (R)=Counterproposing, (I)=AcquiringInfo, (W)=Waiting

Strategic Actions: Propose, Defend, Compromise, Agree, Dissent, Critique, Counterpropose, Wait, AcquireInfo.

D-1: Designer-1, D-2: Designer-2

Figure 3: Speech-Act & Argumentation-Based Negotiation Protocol

As shown in Figure 3, our protocol is composed of sets of speech-acts, negotiation states, and strategic actions. The speech-acts serve as performatives for designers to communicate their claims and arguments. Argumentation during negotiation is carried out through supplying Data & Warrant for Defend and Critique.

In Figure 3, negotiation starts when D-1 proposes a claim, e.g., “Hinge position should be made $20cm<h_g<25cm$.” Upon evaluating this claim, D-2 may decide to Agree or Dissent. He/She may also request Data and/or Warrant, through Critique, to further understand the claim. Being dissented or requested, D-2 supplies Data and Warrant to Defend his/her claim or stance. Still not satisfied by the Data and Warrant, D-2 can Counterpropose a new claim, e.g., “Hinge position should be made $25cm<h_g<30cm$.” Upon evaluating this new claim, D-1 may Agree, Compromise (i.e., modify D-1’s stance), or Dissent, or start it over again by re-Propose a new claim. During this negotiation process, a designer may choose to Acquire more information including engaging in negotiation with a third party or just Wait. Higher level issues used in Data & Warrant for Defend and Critique can be elevated into Claims in Counterpropose or re-Propose actions.

3.4 Negotiation Strategies and Tactics

Given the model of issues, arguments, and the protocol described above, the effectiveness of negotiation depends on how the participating designers decide on strategic actions and how they compose proposals and arguments. The former is related to negotiation strategies and the latter to tactics. Strategies and tactics together determine the direction of negotiation: whether to explore the value space of the current issue, or identify new issues at the same level, or to move to a higher level of relevant issues.

Both strategies and tactics are often domain dependent and sometimes even task dependent. In our research on automotive design, we developed following strategies.

- Value exploration (VE): Try to stick to the current issue and explore its value space extensively.
- Issue exploration (IE): Try to move to, or create, new issues at the same level to avoid conflicts.
- Function exploration (FE): Try to move up to function requirement level to resolve function conflicts.
- Criteria exploration (CE): Try to move up to component or system criteria level and redefine them.

Each strategy is composed of a set of IF-THEN rules with the tendency of moving negotiation to a specific direction. Strategies VE and IE are more effective for routine design tasks, while FE and CE for non-routine ones.

3.5 Roles of Agents

In ANF, agents play an important role in assisting human designers and computer systems to carry out negotiation. Each designer and computer system has an assistant agent that handles all communication processes and facilitates the application of negotiation protocol and strategies. Following is a list of major roles of agents.

- Track issue dependencies between designers and design systems.
- Identify issue discrepancies and notify designer.
- Provide proposal preparation templates.
- Facilitate negotiation communications.
- Suggest strategic actions based on selected strategy.
- Track all negotiation processes and commitments.
- Report statistics of negotiation processes.

When an agent is assisting a computer system, then it will carry out the negotiation autonomously based on pre-defined negotiation rule sets.

4 A CASE EXAMPLE

The negotiation framework described above has been implemented based on a Java-based intelligent agent system developed in our previous research [8]. Prototype systems have been developed to solve simple machine layout design problems and more realistic automotive front door design problems. Due to the space limitation, we skip the implementation details and discuss negotiation scenarios of a simple machine layout design problem, shown in Figure 4.

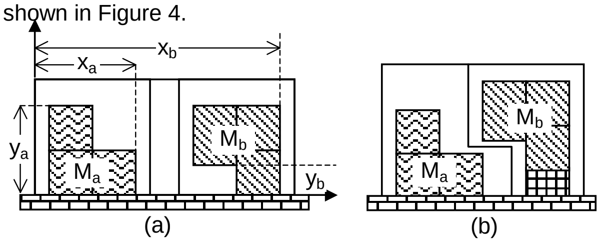


Figure 4: A Layout Design Case Example

In Figure 4(a), designers Da and Db are responsible for compartment layout design for machines M_a and M_b, respectively. Because a new machine will be placed right beside M_b, it must be moved to the left so that Xb < 3m, leaving enough space for the new machine. Figure 5 shows a close-to-natural-language version of the negotiation scenarios between the agents of Da and Db.

This example involves following issues:

Parameter value:	`[Xb=?]`
Parametric constraint:	`[VentilationConstraint]`
Parameters:	`[Xb]` and `[Yb]`
Function Requirement:	`[Ventilation]`

```
Db: Propose:  [Xb IS <3m]
Da: Critique: NOT [Xb IS <3m]
       SINCE (Need more info).
Db: Defend:   [Xb IS <3m]
       SINCE (New Mc moves in)
Scenario 1: Value exploration
Da: Cpropose: [Xb IS 3.5m]
       SINCE [VentilationConstraint]
Db: Compromise: [Xb IS 3.25m]
       SINCE IF [Xb HAS 3.25m]
        THEN [VentilationConstraint] (ok)
Da: Agree.
Scenario 2: Issue exploration
Da: Cpropose: [Xb IS 3m]&[Yb IS >1.2m]
Db: Critique: NOT [Yb IS >1.2m]
       SINCE (Need more info)
Da: Defend: [Xb IS 3m] & [Yb IS >1.2m]
       SINCE IF [Xb IS 3m]&[Yb IS >1.2m]
        Then [VentilationConstraint] (ok)
Db: Critique: NOT [Yb IS >1.2m]
       SINCE (Need Approval)
.. .. ..
Db: Agree: [Yb IS >1]
       SINCE (Approval obtained)
Scenario 3: Function exploration
Da: Cpropose: [Xb IS 3m]&[NewVentilation]
Db: Critique: NOT [NewVentilation]
        SINCE (Need more info)
Da: Defend: [NewVentilation]
        SINCE (Natural ventilation cannot
        satisfy [VentilationConstraint])
Db: Cpropose:
        [NewVentilation]&[AirConditioner]
... .. ..
                (Cpropose = Counterpropose)
```

Figure 5: Negotiation Scenarios of A Case Example

In scenario 1, the negotiation is focused on finding a common ground within the current issue. Designers give-and-take and finally reach an agreement. In “issue exploration” scenario, Da brought up a new issue Yb, and by counter-proposing “raising machine Mb by 1.2m.” This way, shown in Figure 4(b), the [VentilationConstraint] will be satisfied and the original issue Xb=3m can be agreed. After receiving the approval, Db agreed. In “Function exploration” scenario, Da did not stay at the levels of parameter values and parameters. Instead Da looked into the functional reason behind “keeping distance between the machines”. It was for “allow natural ventilation.” Da then counter-proposed [NewVentilation] as a solution to resolve the original conflict. This move of level of issue led to negotiation about “how to design the new ventilation.” Db proposed to use “air conditioner” and the negotiation continues until final agreement can be reached.

From this example, it can be seen that the proposed negotiation protocol can accommodate different strategies and is flexible to deal with ill-structured problems as well as well-structured problems. The multi-level issue and argument model allows designers to move negotiation into different directions guided by the strategies adopted. The communications between designers Da and Db are totally facilitated by their associated agents. The agents not only provide protocol templates but also search for Issues, Data, and Warrants (constraints) to fill in the blanks.

Our experience with ANF thus far has shown that having a sophisticated model to represent and capture design context including design Issues, Arguments, constraints, dependencies, and design decisions is of key importance for successful agent-based negotiation support. The richness of such a design context model allows agents to be more “intelligent” and provide sophisticated assistance for designers.

5 CONCLUDING REMARKS

Complex engineering problems always involve multiple criteria and multiple designers. Hence negotiation is inherent in engineering design. Design negotiation is not merely for resolving conflicts. It is a means for designers to explore new design spaces, seek new opportunities and arrive at innovative designs. In this paper we discussed the issues involved in design negotiation, and presented protocols, information models and an agent-based system to support engineering negotiation. The case example demonstrated the efficacy of our framework and explicated the important roles of negotiation in engineering design. Our current research focuses on developing a more sophisticated design context model, linking the framework with decision theoretic methods, and applying the framework to more realistic engineering design problems.

6 REFERENCES

[1] CIRP ECN (Engineering as Collaborative Negotiation) Website. http://mspde.usc.edu/ecn.

[2] Pruitt, D.G., 1981, Negotiation Behaviour—Organizational and Occupational Psychology, Academic Press, New York

[3] Gulliver, P.H. 1979. Disputes and Negotiations: A Cross-Cultural Perspective. Academic Press, New York.

[4] Toulmins, S.E. 1969. The Uses of Argument. Cambridge University Press, Cambridge, UK.

[5] Sycara, K.P. 1989. Multiagent Compromise via Negotiation, In Distributed Artificial Intelligence, Gasser, L. and Huhns M. Eds. Vol. 2. Pitman Publishing, London, 119-137.

[6] Parsons, S. Sierra, C. Jennings, N.R. 1998. Agents that reason and negotiate by arguing, in Journal of Logic and Computation, 8/3:261-292.

[7] Raiffa, H. 2003. Negotiation Analysis. Belknap Press.

[8] Jin, Y. and Lu, S. 1998. An Agent-Supported Approach to Collaborative Design, CIRP Annals, 47/1:107-110