NEGOTIATION DESIGN PATTERN FOR SERIOUS GAMES
MARTIN RUSKOV, PETKO RUSKOV

Sofia University, Faculty of mathematics and Informatics

martin.ruskov@gmx.net
petkor@fmi.uni-sofia.bg

ABSTRACT

Purpose – The purpose of the present paper is to propose a pattern towards filling the gap between demand for serious games on negotiations and lagging supply. A solution for dialogue-based negotiation games is described, to be defined by a human tutor and used in a virtual business environment.

Design/methodology/approach – At first state-of-the-art game design patterns and e-Business process patterns are identified. Then following the genesis of a pattern, a design pattern for negotiation games is analyzed, described, tested and defined, based on mathematical game theory.

Findings – Since context of training plays a critical role in the negotiation games, precise tutor scenarios need to be customizable to fit the organizational needs of education and training. On the basis of our research, a negotiation pattern is clearly defined and preliminary tested.

Research limitations/implications – The result needs to comply with the limitations of current dialogue-based systems.

Practical implications – The adoption of the concept will give business stakeholders a hands-on approach to learn, test and approve negotiation skills.

Originality/value – This paper provides an original pattern for serious games about dialogue-based negotiation.

Keywords: serious games, design patterns, negotiation analysis

INTRODUCTION

For many centuries negotiation has been the way people find mutually profitable solutions. With recent trends in globalization and mixture of cultures, decentralization of decision-taking in organizations, digitalization and information society, and orientation of economies toward interests of individuals and customers, the science and art of negotiation becomes ever more important (Nash, 1950; Kersten, 2001; Raiffa, 2002)

Dialogue is the natural way people communicate, and the dialogue management systems have reached a stage where they are able to take part in consistent bilateral conversations (Churcher, 1997; Ruskov, 2005). Mathematical game theory has tried to formalize the negotiation process. It has reached a situation where there is no unique solution to distribute common net profit. People can get education in this area only by specialized university courses or literature.
Chris Crawford’s idea that “games are the original educational technology, the natural one, having received the seal of approval of natural selection” (Crawford, 1982) recently has again attracted attention of business, science and education communities. Nowadays, computer games have become a very popular form of education, because of the fun they bring into the process. On the other hand they have become very important industry, which apply amount of methods from software development. One of them - the use of patterns in game design is in its early years (Alexander, 1977; McNaughton., 2004; Björk, 2006). Design patterns are also frequently used in business (IBM 2006). Like any semi-formal method, patterns are useful as reasonable efforts to memorize and re-apply them in adequate way. The use of game design patterns can be divided into two categories: analysis and design. Analysis requires an existing game and study what game design patterns exist. Design patterns pass on to the creation of an idea, or description of a game element, or of formalizing a game idea or concept into a more structured description.

The negotiation analysis in games has been studied in a certain perspective by researchers, but there is no pattern about it (Meijer, 2005). Yishay Mor and his team describe a deep literature review about the process of designing and developing mathematical games (Mor, 2006). All these lead us to research a novel game design pattern, which describes how the art and science of conversational negotiations could be practiced in a computer game using artificial agents, based on mathematical models. One of the difficulties is to make the communication between players and agent possible. This paper describes the internal model that an artificial conversational agent could use in order to be able to verbally (using words) negotiate in a predefined scenario. This paper was supported by the project PRIME (Contract no.: FP6-IST-016542) - Providing Real Integration in Multi-disciplinary Environments whose goal is to give business professionals a learning environment where they can experiment with new ideas and learn how to handle the entire life cycle of products and processes for all stakeholders of the organization.

**PROBLEM DEFINITION**

We introduce a situation where a mentor or trainer establishes the business environment, creates the game scenario and supervises the players. In our case bilateral verbal negotiations are addressed. A mentor verbally defines a scenario to the human player and its formalization to the system. The mentor is responsible for defining a scenario, for bringing the player to collaborate in reaching a solution and for discussing passed negotiations in order to draw conclusions. In this way the pattern, defined by the elements in this paper, becomes a tool in a larger educational framework. The scenario itself consists of the following components:

- A verbal scenario description for the player
- Dimensions with their names and ranges
- Utility function and willingness to cooperate of the artificial intelligence (AI) agent
- Environment constraints

Formally there are several parameters of a deal. During the negotiation process new parameters could emerge and old ones could become irrelevant. We make the assumption that the background cognitive processes are too complex for a computer to emulate and therefore we leave these fixed. As commonly done in mathematical game theory, we consider these parameters to be the dimensions of the negotiation. For technical purposes, let us assume that these dimensions can be mapped to numerical values. While this is straightforward for material, cost and time bargaining, there are other features, such as scope, whose transformation into a numerical space is not that simple. We believe that with some modeling this could be managed and leave this responsibility to the mentor. Very often dimensions of a negotiation are not mutually independent. Since, again, cognitive processes to
comprehend the exact dependencies are too difficult to be implemented in general, again it is expected that the mentor defines the interdependencies in the scenarios.

We illustrate our approach with an example scenario of buying a laptop computer. In reality, people rarely negotiate for single PC’s without external motivation, but it is simple enough to be understandable and rich enough to serve as an example. The exact formulation of dimensions and conditions depend on the exact situation. Example scenario definition and a sample game are given in the appendices 1 and 2.

When considering the problem, we assume that the human player (called simply player or she) has interest in finding a solution and that she will be ready to react on some of the offers. Since the game is supervised by the mentor (also referred as he), we assume that the player too will put an effort in overcoming deadlocks. So the goal of the game is not to find an equilibrium, but how to make it as coherent as possible to the predefined preferences. The assessment of the outcome is again left to the mentor.

GAME FRAMEWORK

For a game to be attractive to the player, it needs to be made as accessible as possible. In order to prevent the player from trying to find flaws in the game, we conceal whether the game opponent is a human or an artificial agent. This is to be achieved by integrating the negotiation rounds in one massive-multiplayer online game with additional AI agents. In each round a player is presented a new scenario with a new game opponent (figure 1).

![Figure 1 Game flow](image)

Although some variations of game-theoretical negotiation modeling are applied both in computer programs and by experienced businessmen, propositions and decision outcomes are difficult to be communicated between different parties. While machines “think” in numerical constraints, utilization and coefficients, people usually consider these more informally and intuitively, with experience substituting formalisms. In order to make communication accessible to a non-professional (e.g.
someone who is willing to become one), mathematical models and human thinking have to be brought to the same medium.

This communication problem could be solved by approaches, commonly used in practice. In accordance with recent trends, we have put our effort to formalize these approaches as a flexible and reusable software design pattern for game-based learning. It is a sub-pattern for the communication between AI agent and human player in the game flow. Since users are not expected to follow a structured negotiation process, the agent will try to impose one. ebXML presents a process formalization that could be used for the purpose (figure 2), (ebXML, 2001). This process has the advantage that even if users attempt to deviate from it, the agent could easily bring the conversation back on track by just trying to identify its progress and keep talking according to the scheme. Hamza and Fayad introduce a negotiation analysis pattern that aims to formalize the negotiation environment (Hamza, 2004). For the purposes of this paper, however, we put the focus on the dialogue between the parties, rather than the framework they could be put into. Chang and Woo have developed a speech-act-based negotiation protocol for distributed AI systems in a paper that uses similar concepts, but for different purposes (Chang, 1994).

![Figure 2 A negotiation process model](image)
The theory of speech acts as developed by John Searle (Searle, 1969) and further used in many contemporary natural language understanding systems classifies user utterances into speech act categories. Together with some semantic information this could easily be translated into a numerical constraint language. This technique emerges ever more often as common practice in dialogue-based interaction. Its popularity and widespread use give us motivation to define it as a software design pattern – the speech act classification pattern (figure 3). It is a layer between user input and its interpretation by the system, that maps free-form human speech to the meaning it could have in the context of the application and vice versa.

In games that use the pattern, user utterances are reduced to only partial understanding of pragmatics and semantics expressed. Any other information by the user could be considered off-topic. In such cases the agent may “act dumb” and ignore the utterance or just ask for clarification. Appendix 3 shows some examples how free sentences could be mapped to categories and numerical information. The distinction between speech acts is based on the exact phrases by the user. Utterings that express more than one speech act are more complicated to handle, but techniques like sentence breakdown allow these to be handled separately. Existing systems have shown that this approach enables intuitive interaction between player and agent. Recent applications of this sub-pattern easily fall into conversational “traps” and so become distinguishable from a real person’s behavior. Practice has shown that if the human correspondent is good-intended, a successful conversation could be led. However the success depends on the effort put in classification and the set of predefined understandable phrases (Ruskov, 2005).

The inverse process of presenting AI responses to the user is simpler. In this case speech acts, accompanied with their semantic information are expanded into sentences in a random manner.

GAME-THEORETICAL MODELING

Negotiation analysis, a field of game theory, will be used to model the reasoning process of the agent in the negotiation pattern. The problem definition fits in the classical framework of two-person cooperative games (Nash, 1953). Research made by Gregory Kersten (Kersten, 2001) on general modeling of such games, however for the purposes of education, a comprehensible simplification is more appropriate.

Our pattern focuses on finding an equilibrium in N dimensional space with linear modeling. The utility functions of the agent and of the player (verbally defined) are not necessarily the same, so this is a non-zero sum game. The agent should play an optimal strategy, so balanced equilibria are the best the player could get. The information the agent extracts from the negotiation dialogue is about restricting the set of feasible solutions. Without any information exchange the whole solution space represents possible negotiation outcomes. Interaction between negotiators defines additional constraints to those, imposed by the mentor. These constraints and the utility function of the agent, given by the mentor,
define the problem as an optimization problem. We will consider it statically for the situations between interactions with the player. However, since negotiation is an iterative process, the goal of the agent at one step will be not to propose an optimal solution, but to lead the negotiation in the direction where its utility function would gain most. The motivation is that in a real business often people tend to make linear approximations, which are good enough and easier to handle. Furthermore, in the case of linear modeling, this local approach delivers a globally optimal solution (Dantzig, 1998).

In order to be able to propose solutions that are more likely to be accepted by the game opponent, it is helpful to consider what is known about player’s utility too with an appropriate coefficient. This coefficient represents the willingness of the agent to cooperate. Zero means the agent does not care about the interests of the player and value one that common interest is equally important to its personal. The only way the agent could gather information about player’s (expectedly implicit) utility is to interpret her dialogue contributions and try to construct an approximation (Utility_{player}). As a result, the agent will not propose according to its utility function (Utility_{agent}) as it is defined, but according to the difference in the formula:

\[
\text{Utility}_{\text{total}} = \text{Utility}_{\text{agent}} + \text{Coefficient} \times \text{Utility}_{\text{player}}
\]

The two-dimensional cuts of the situations after speech act 6 (price and delivery time shown) and speech act 13 (only dimensions price and OS shown) form the example in Appendix 2 are shown in fugure 4.

Figure 4 Negotiation-theoretic models of two negotiation situations

In the situation at right there is a non-empty feasible set (the gray area). The player proposes an equilibrium that is convenient for the agent, so it accepts. Note that the mentor-imposed constraint that delivery time should not exceed 2 days is illustrated by the black vertical line at right. In the left situation, there is conflict of interests regarding the operating system dimension of the example. The agent is told that for the player operating system is more important than guarantee. However it is in its
own interest to sacrifice rather in price, than in guarantee (reducing the guarantee would almost not increase its utility). This is why it makes the suggestion in speech act 14. The described situations present only two from many possible configurations, but they illustrate well the decision-taking process of the agent in general.

The presented mathematical model shows how agents can fill the gap of reasoning in the general negotiations design pattern.

FUTURE WORK

A software prototype of a game with the proposed pattern needs to be created to be able to get empirical data on its usability. Once this pattern shows its value, it could be developed further to integrate with some of the works described in the following two works.

First, Chang and Woo use for their purposes a speech act classification by Ballmer and Brennenstuhl (Chang, 1994). It needs to be investigated how this relates to the simplistic classification in Appendix 3 and whether it could serve for extension of the speech act classes set.

Second, the strategy determination process could be further developed by introducing market and using Sackmann’s models as price norms. Stafan Sackmann has proposed a mathematical model to determine prices in bilateral negotiations, according to current market performance, which is a useful related consideration, but beyond the scope of this paper (Sackmann, 2004).

CONCLUSION

In this paper, we have shown how patterns design method offer a flexible and creative way for the different levels of abstraction and match the varying needs of an increasingly complex negotiation in serious game development process. Implementing negotiation design patterns in computer games will permit the mentor to supervise what will occur in the game during the negotiation process.

The intended player audience for serious business games that could use this pattern is business staff like sales, marketing and procurement personnel, managers or management students without theoretical background in negotiation theory or other related mathematical theories. The purpose of the game pattern, presented in this paper is to give players practical experience with negotiating in a situation where optimal strategies are used. This negotiation design pattern for AI agents also enable the creation of various games using dialogues in game-based education and training programs for people in different positions. The adoption of the proposed concept will give mentors, trainers and business stakeholders a hands-on game based approach to learn, test and approve negotiation skills of there’s staff.

References
Dantzig, G. (1998), Linear Programming and Extensions, Princeton University Press


Meijer, S., Zuniga-Arias, G., (2005), Experiences with the Mango chain game, Proceedings of the 9th Workshop of the IFIP WG 5.7 Special Interest Group on Experimental Interactive Learning in Industrial Management, June 5-7, Espoo, Finland, pp. 123-132.


Appendix 1: Scenario Example

Verbal scenario description for the player:
A software company has given one of its developers with $3000 to buy a laptop for his work. He doesn’t want to install the operating system himself, but the company could to this for him at the cost of $100. There is a government legislation that forces suppliers to deliver computer equipment within two days. The developer goes to a computer shop and needs to bargain about the computer he gets.

Dimensions with their names and ranges
price (cash amount – positive integer)
operating system (true, false)
guarantee (time period, years)
delivery time (time period, hours)

Utility function of the artificial intelligence (AI) player
5 * price - guarantee + 3 * delivery time – 100 * operating system
Appendix 2: Game Example

U: [GREETING] Hello!
A: [GREETING] Welcome!
U: [INTEREST] I need a laptop
U: [CONSTRAINT] and I have $3000 for it.
A: [OFFER] We have one for $2000,
A: [CONSTRAINT] I can deliver it no less than 8 hours.
U: [OFFER] Can you deliver it by tomorrow?
A: [AGREEMENT] Yes.
U: [INTEREST] Is there operating system included?
A: [CONSTRAINT] No OS.
A: [OFFER] but there is three years guarantee
U: [INTEREST] I don't need guarantee,
U: [INTEREST] I want OS
A: [OFFER] You could have the laptop for $1800,
A: [CONSTRAINT] but no OS.
U: [INTEREST] And the guarantee?
A: [CONFIRMATION] And three years guarantee.
U: [AGREEMENT] Ok.
A: [OFFER] Here is the offer: price: $1800, no OS, three years guarantee, delivery in 24 hours.
U: [AGREEMENT] I agree.
A: [PARTING] Thank you and please come again
U: [PARTING] Good bye.

Appendix 3: Examples of utterance to speech act mapping and information extraction

A brief explanation of sample speech acts and several examples follows.

- OFFER represents an offer or new position that the player declares
- INTEREST is mentioning of a dimension that is important (or unimportant) to the player, i.e. she displays interest to gain (give) in this direction
- CONSTRAINT a limitation of the bargain, that the player introduces
- AGREEMENT acceptance of OFFER
- GREETING opening act, common for interpersonal dialogues
- PARTING closing act, similar to GREETING
- CONFIRMATION similar to OFFER, but does not present any new information, only verifies something already mentioned
<table>
<thead>
<tr>
<th>Sentences</th>
<th>Speech Act</th>
<th>Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>I give you $10000 under these conditions</td>
<td>OFFER</td>
<td>$10000 price</td>
</tr>
<tr>
<td>I want $500!</td>
<td>OFFER</td>
<td>$500 price</td>
</tr>
<tr>
<td>The contract comes with 3 year guarantee and lifelong service.</td>
<td>OFFER</td>
<td>3 years guarantee and infinite service</td>
</tr>
<tr>
<td>Would you make a discount if I bought 20 pieces?</td>
<td>INTEREST</td>
<td>Price/20 quantity</td>
</tr>
<tr>
<td>I have only $10</td>
<td>CONSTRAINT</td>
<td>Less than $10 price</td>
</tr>
<tr>
<td>There is no way that I accept a delivery after a week</td>
<td>CONSTRAINT</td>
<td>Less than 1 week delivery time</td>
</tr>
<tr>
<td>I need it cheaper</td>
<td>INTEREST</td>
<td>Price</td>
</tr>
<tr>
<td>I do not have enough money</td>
<td>INTEREST</td>
<td>Price</td>
</tr>
<tr>
<td>I don't care about the color!</td>
<td>INTEREST</td>
<td>Not color</td>
</tr>
</tbody>
</table>