Towards Automated Negotiation Agents that use Chat Interface*

Inon Zuckerman[1],Erel Segal-Halevi [2], Sarit Kraus[2,3], Avi Rosenfeld[4]
[1] Department of Industrial Engineering and Management,
Ariel University of Samaria, Ariel, Israel 4070000
[2]Department of Computer Science
Bar-Ilan University, Ramat-Gan, Israel 5290002
[3] Institute for Advanced Computer Studies
University of Maryland, College Park, USA 20742
[4]Department of Industrial Engineering
Jerusalem College of Technology, Jerusalem, Israel 9116001
inonzu@ariel.ac.il, erelsgl@gmail.com, sarit@cs.biu.ac.il, rosenfa@jct.ac.il,

ABSTRACT

To date, a variety of automated negotiation agents have been created. While each of these agents has been shown to be effective in negotiating with people in specific environments, they lack natural language processing (NLP) methods required to enable real-world types of interactions. In this paper we study how existing agents must be modified to address this limitation. After performing an extensive study of agents' negotiation with human subjects, we found that simply modifying existing agents to include an NLP module is insufficient to create these agents. Instead the agents' strategies must be modified to address partial agreements and issue-by-issue interactions.

1. INTRODUCTION

Negotiation is a basic task that forms a basic element in our daily lives. We often find ourselves in situations, whether simple or complex, that require negotiations. Most negotiations are mundane, such as haggling over a price in the market, deciding on a meeting time, or even convincing our children to eat their vegetables. However, they can also have colossal effects on the lives of millions, such as negotiations involving inter-country disputes and nuclear disarmament [9].

To date, a variety of agents have been created to negotiate with people within a large spectrum of settings including: the number of parties, the number of interactions, and the number of issues to be negotiated. Katz and Kraus [11] proposed an agent for one-shot interactions in an environment where only one issue needed to be negotiated between two parties (bilateral negotiation). The *AutONA* agent was developed for repeated interactions between buyers and sellers over the price and quantity of a given product [3]. More complex agents have been created for multilateral negotiations involved several issues to be considered. For example, the *KBAgent* has been shown to be the most effective agent in achieving agreements with people in several domains involving multiple attributes [19].

Two key common elements exist throughout all of these previous agents. First, these agents are all based on the assumption that the human negotiators use bounded rationality. People did not successfully reach agreements with agents based on notions of equilibrium or optimal methods, and thus alternatives needed to be found for all agents [15]. Second, all agents needed mechanisms for dealing with incomplete information. This is typically done through reasoning about the negotiating partners by learning their preferences and strategies [8].

The key point this paper addresses is a study of how to extend current state of the art agents to use natural language processing. Unfortunately, this ability is lacking to current state of the art negotiation agents – something that has been previously noted [15]. This inherent limitation requires these agents to "force" their human counterparts to interact via menus or other non-natural interfaces.

Towards creating agents that use natural language, this paper addresses what extensions, if any, are needed to bridge this gap. As a first step towards creating negotiation agents with full NLP capabilities, we conducted extensive studies of interactions between the leading automated negotiation agent and people. We compared how people negotiated with this agent through its previous menu-based interface, and a new chat-based interface that allowed people to converse freely with the agent.

This paper presents two important results based on this study. First, we discovered that the automated negotiation strategies did not transfer well to more natural forms of conversation. Simply adding a chat-based interface instead of a menu-based interface to the existing agent yielded agreements that were significantly *worse* for the agent, while the utility for the human player remained the same. In addition, we found that the human partners were significantly happier with the final agreement, and they perceived the final outcome to be more balanced if they were using the chat-based interface, despite the fact that they attained the *same* average utility in both interfaces.

Second, we managed to isolate the reason for the algorithm's inability to cope with partial agreements as the main cause for its decreased performance. One key issue that we study is the centrality

*This work was supported in part by ERC grant #267523, MURI grant number W911NF-08-1-0144 and the Google Inter-University Center for Electronic Markets and Auctions.

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.*

of creating partial agreements within natural language based negotiation agents. It is known that bounded rational agents (such as humans) find that simultaneously negotiating a complete package might be too complex [1, 2], and therefore they prefer to negotiate issue-by-issue. As our next Section details, this is an open issue within the general negotiation research community, but is evidently a key issue that must be addressed by agent designers.

2. RELATED WORK

This paper's main contribution lies in empirically analyzing how negotiation agents should be extended to support more natural interfaces, and specifically a chat interface. Extensive studies in the field of Human Computer Interactions (HCI) have noted that the goal of any system should be an intuitive interface with the stress being put on creating agents which operate in environments which are as real and natural as possible [5, 6]. Thus, following these approaches, it is critical to develop natural language support for negotiation agents to allow for these types of "normal" interactions [13]. This form of typing as natural interaction is referred to as *Natural-language interaction* (NLI) in the literature. There have been numerous informal tests of NLI systems, but few controlled experimental comparisons against some other design [21].

While automated negotiation agents have been developed for quite some time, unfortunately, even state of the art negotiation agents do not yet support natural language interactions. Over twenty years ago in [14] they developed an agent called *Diplomat*, that played the Diplomacy game with the goal to win. Byde *et al.* [3] developed *AutONA*, an automated negotiation agent. Their problem domain involves multiple negotiations between buyers and sellers over the price and quantity of a given product. Jonker *et al.* [10] created an agent to handle multi-attribute negotiations which involve incomplete information. The *QOAgent* [17] is a domain independent agent that can negotiate with people in environments of finite horizon bilateral negotiations with incomplete information. The negotiations consider a finite set of multi-attribute issues and time-constraints. We focus on the KBAgent, which like the *QOAgent* also considers negotiations with a finite set of multi-attribute issues and time-constraints, but has been shown to be the most effective agent in achieving agreements with people in several domain [19]. This area continues to be quite popular, with one active research avenue being the ANAC (Automated Negotiating Agents Competition) Workshop. Since 2010, this competition has focused on agents that use the GENIUS interface.[1] However, we note that even to date, this competition focuses on agent-agent interactions and the interface supports only menu-based interactions between agents and people.

To address this limitation, we study what logical extensions are needed, if any, to make existing negotiation agents suitable for natural language. Previously economic and behavior research into people's negotiation would suggest that the current approach of attempting an agreement on all issues simultaneously will not be effective. For example, Bac and Raff [1] found that simultaneously negotiating a complete package might be too complex for individual buyers. Furthermore they show that, in the context of incomplete information with time discount, the more informed player ("strong" in their terminology) will push towards issue-by-issue negotiation. Busch and Horstmann [2] found some people might like to decide all issues at once, while others prefer to decide one by one. Chen [4] studied issue-by-issue negotiation with opt-out factor, and argues that when the opt-out probability is low, agent prefer to negotiate a complete package because intuitively we know that the negotiations can last long enough so that agents can get to a "win-win" situation. However, with high opt-out probability, agents prefer issue-by-issue negotiation. Thus, one key contribution of this paper is its study as to how people react to agents that do not propose issue-by-issue agreements.

[1]The GENIUS and ANAC websites can be reached though http://mmi.tudelft.nl/negotiation/index.php/Genius

3. METHODOLOGY

The main goal of this research was to push the envelope of automated negotiators research by moving from menu-driven interfaces to chat based environments. As this work transitions from the fruitful work of previously developed agents, we intentionally chose to base ourselves on these agents and the complex environments they had studied. Thus, we shied away from dealing with overly simplified settings, such as those with full information, single issues, or alternating turn based offers, and instead considered a complex problem with partial information, multi-attribute negotiations, and an unconstrained interaction protocol. In this section we detail the negotiation problem we considered, the state of the art KBAgent agent we based our study on, and the GENIUS environment used by the agent.

3.1 Problem Description

The negotiation environment we consider can be formally described as follows: We studied *bilateral* negotiation in which two agents negotiate to reach an agreement on conflicting issues. The negotiation can end either when (a) the negotiators reach a full or partial agreement, (b) one of the agents opts out (denoted as OPT), thus forcing the termination of the negotiation with a predefined opt-out outcome, or (c) a time limit is reached, that results in a predefined status-quo outcome (denoted as SQ).

The negotiations resolve around *multi-attribute* characteristics. There is a set of issues, denoted as I, and a finite set of values, O_i for each issue $i \in I$. *Partial* agreements are possible as subset of the issues contains $\perp \in O_i$. An offer is denoted as $\vec{(o)} \in O$, and O is a finite set of values for all issues. The negotiations are sensitive to *time*. Time impacts the utilities of the negotiating parties, and is defined as $Time = \{0, ..., dl\}$, where dl is the deadline limit. Each agent is assigned a time cost which influences its utility as time passes. The time effect may be negative or positive with respect to the utility.

The negotiation *protocol* is fully flexible. As long as the negotiation has not terminated earlier, each side can propose a possible agreement, reject a previously offered agreement, opt-out of the negotiation, or communicate a threat, promise, or any general remark. In contrast to the model of alternating offers [18], each agent can perform up to $M > 0$ interactions with the opponent agent in each time period.

Last, we consider environments with *incomplete information*. That is, agents are not fully aware of the utility structure of their opponents. We assume that there is a finite set of utility structures which will be referred to as agent types. For example, one type might model a long term orientation regarding the final agreement, while another might model a more constrained orientation. Formally, we denote the possible types of the agents $Types = \{1, ..., k\}$. Given $l \in Types$, we refer to the utility of the agent of type l as u_l, and $u_l : \{O \cup \{SQ\} \cup \{OPT\}\} \times Time \rightarrow \mathcal{R}$. Each agent is fully aware of its own utility function, but it does not know the exact type of its negotiating partner.

3.2 The KBAgent

The state-of-the-art automated negotiator for the above environment is the *KBAgent* [19]. It has been shown that the *KBAgent* negotiates efficiently with people and achieves better utility values than other automated negotiators. Moreover, the *KBAgent* achieves significantly better agreements, in terms of individual utility, than the human counterparts playing the same role.

The main difference between the *KBAgent* and other agents is its inherent design, which builds a general opponent model. *KBAgent* utilizes past negotiation sessions of other agents as a knowledge base for the extraction of the likelihood of acceptance and offers which will be proposed by the other party. That data is used to determine which offers to propose and what offers to accept. One of its main advantages is that it can also work well with small databases of training data from previous negotiation sessions.

In order to generate an offer, the *KBAgent* creates a list ordered by the QOValue, which is an alternative to the Nash bargaining solution (see [17] for exact definition). The first offer that is proposed is the one with the maximal QOValue. The other offers are picked from the ordered list based on the concession rate the *KBAgent* applies and are chosen with a decreasing QOValue for the agent and an increasing utility value for the other party. To decide which offers to accept, the *KBAgent* determines a time dependent threshold to decide whether to accept or reject an offer. In order to decide on the optimal threshold, the probabilities learned from the database of past negotiations are used.

3.3 The GENIUS Environment

When conducting computer based experiments with human participants, the interface design might have significant impact on the results as different design decisions might affect the subjects' behaviors [21]. For example, items placed at the top of a drop-down list have higher probability of being selected, default values might have framing effects, etc. Over the years, research on human-agent negotiation suffered from a comparative weakness when a new algorithm was compared to an old one but on a different interface.

To remedy this, an open source negotiation environment by the name of GENIUS was published in order to facilitate research on bilateral, multi-issue negotiations [16]. The environment can be used both as an API for the development and testing of automated agents, and as a simulated environment to run tournaments and experiments in various negotiation scenarios Moreover, since 2010 it has been used as the main tool of a novel annual Automated Negotiating Agent Competition (ANAC).

The front-end interface for human based negotiation experiments is a dialog based graphical user interface. It contains various action buttons, pull-down boxes to select values for issues, and text areas to display information. See Figure 1 for an example. We have used exactly the same interface in terms of its look&feel, but replaced the menus with a single text box for the chat area that will be used to pass messages between the negotiating parties.

3.4 Wizard of Oz

The main goal of our research is to understand whether the constraints of the menu-based interface affects the nature of agreements produced by a state-of-the-art negotiating agent. Stated differently, *we would like to check whether an automated agent developed in a menu-based negotiation environment, will be as effective in a chat-based environment.* Intuitively, it is not easy to say if there is any relationship between the negotiation interface and the negotiation algorithm that is used by the agent. But if such relationship does exist, it should be analyzed so that a new generation of negotiation strategies should incorporate these findings.

In order to study this point, we needed to translate each natural language sentence written in the chat box to an action object that can be accepted by agent. For example:

```
``I offer you a salary of 12,000''
```

should be translated to an object of the form:

```
Offer(Salary=12000)
```

Sentences in natural language might be *ambiguous*. For example, the sentence:

```
``Can you agree to work for 12,000?''
```

can be interpreted in at least two different ways. The first interpretation is a simple query to gather information regarding whether the candidate will agree to work for that salary. An answer to that query will reflect the willingness to accept such a value for that issue in a future agreement:

```
Query(Salary=12000)
```

A second possible interpretation is an offer of that salary. In this case, the person is in fact proposing an offer for this issue, expects a response to this offer, and in fact wishes to conclude a partial agreement:

```
Offer(Salary=12000)
```

The problem of *ambiguity* within natural language is a well known challenge within the field of Natural Language Processing (e.g. [20]). Unfortunately, even state of the art approaches cannot deal with such ambiguity with absolute certainty. While in the future we hope to develop NLP algorithms for negotiation chat agents, as a first step we sidestepped this problem by having people manually decode ambiguity in other people's chat statements. To do so we used the *Wizard of Oz* (WOZ) approach [12, 7].

In WOZ experiments, the users believe that they are interacting with an automated agent directly, but behind the scenes there is a human being that translates their messages to the language that the agent understands. For instance, given the above sentence "Can you agree to work for 12,000?", a human "Wizard" decides which of the possible interpretations is more likely, and sends the correct interpretation to the agent.

An advantage of the WOZ approach is that it allows us to separate the NLP component of the agent from its strategy, allowing us to focus on the negotiation algorithm and proceed to study the question at hand. A snapshot of the WOZ interface used in our experiments can be seen in Figure 3.

4. EXPERIMENTS

In order to properly evaluate the influence of natural language input on automated negotiation agents, we intentionally picked the *job candidate* domain used in previous research [17, 19]. In this domain, a negotiation takes place after a successful job interview between an employer and a job candidate. In the negotiation both the employer and the job candidate wish to formalize the hiring terms and conditions of the applicant. Below are the issues under negotiation:

Salary This issue dictates the total net salary the applicant will receive per month. The possible values are {7000, 12000, 20000}.

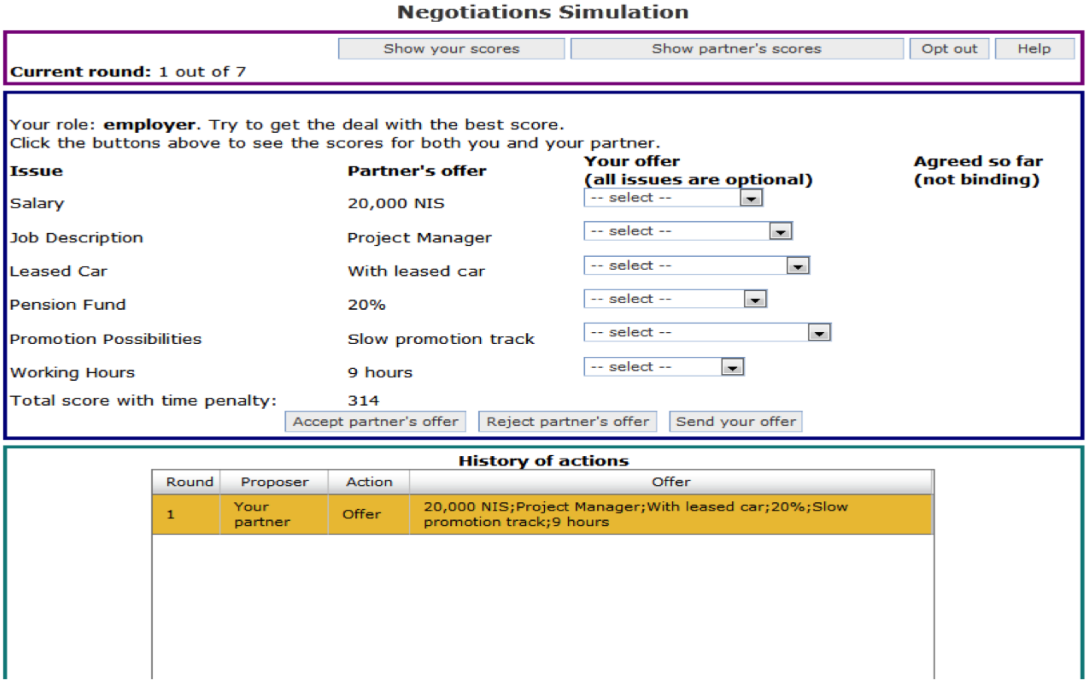

Figure 1: GENIUS menu-based interface

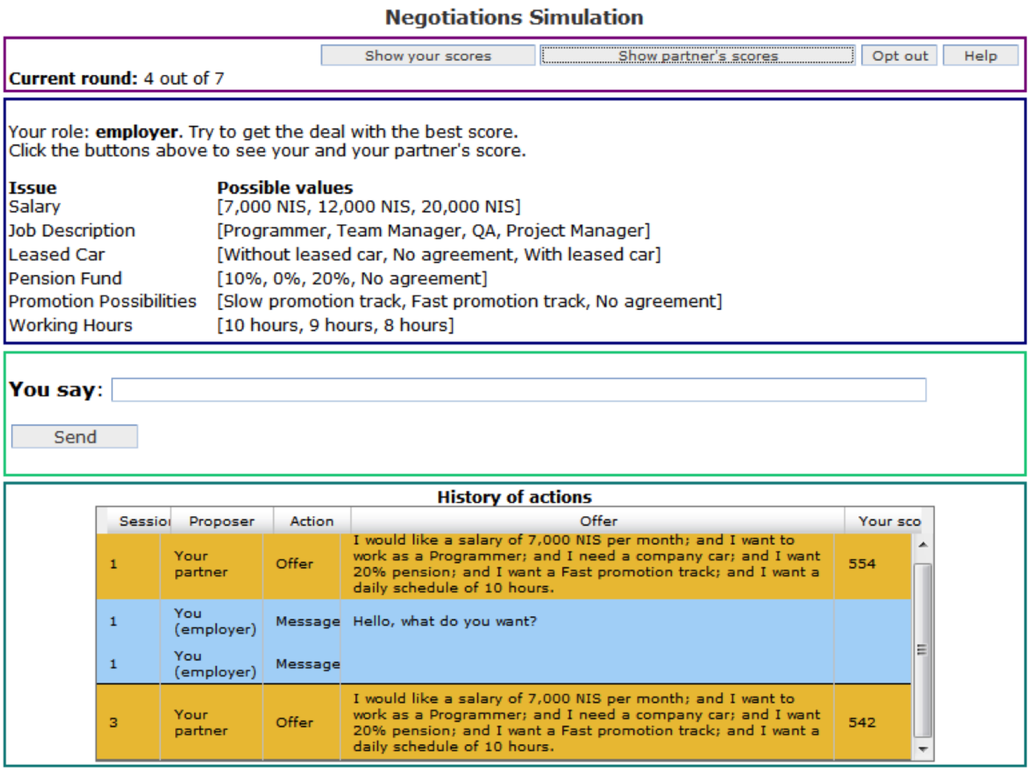

Figure 2: GENIUS chat-based interface

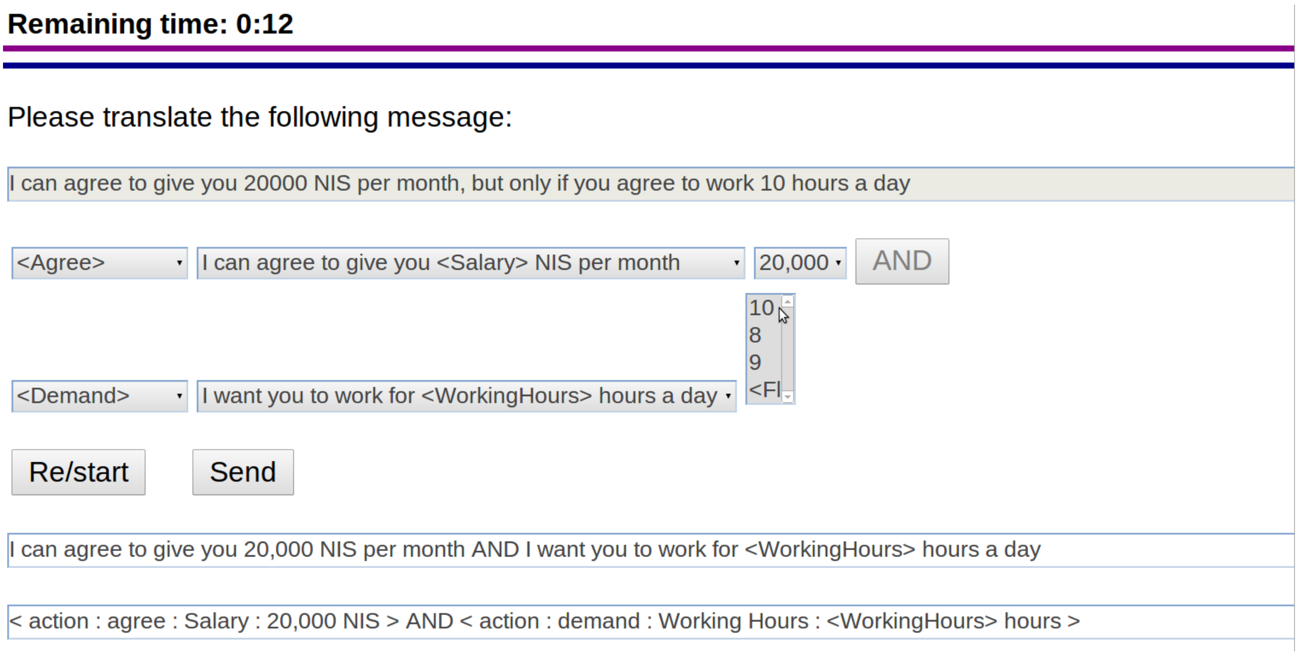

Figure 3: The WOZ interface

Job description This issue describes the job description and responsibilities given to the job applicant. The possible values are {QA, programmer, team manager, project manager}.

Social benefits The social benefits are divided into two categories: company car and the percentage of the salary allocated, by the employer, to the candidate's pension funds. The possible values for a company car are {leased car, no leased car, no agreement}. The possible value for the percentage of the salary deposited in pension funds are {0%, 10%, 20%, no agreement}.

Promotion possibilities This issue describes the commitment by the employer regarding the fast track for promotion for the job candidate. The possible values are {fast promotion track (2 years), slow promotion track (4 years), no agreement}

Working hours This issue describes the number of working hours required by the employee per day (not including over-time). The possible values are {8 hours, 9 hours, 10 hours}.

In this scenario, a total of 1296 possible agreements exist ($3 \times 4 \times 12 \times \ 3 \times 3 = 1296$). Each turn in the scenario equates to two minutes of the negotiation, and the negotiation is limited to 30 minutes. If the sides do not reach an agreement by the end of the allocated time, the job interview ends with the candidate being hired with a standard contract, which cannot be renegotiated during the first year. This outcome is modeled for both agents as the status quo outcome. Each side can also opt-out of the negotiation if it feels that the prospects of reaching an agreement with the opponent are slim and it is impossible to negotiate anymore. Opting out by the employer entails the postponement of the project for which the candidate was interviewing, with the possible prospect of its cancelation and a considerable amount of expenses. Opting-out by the job candidate will make it very difficult for him to find another job, as the employer will spread his/her negative impression of the candidate to other CEOs of large companies. Time also has an impact on the negotiation. As time advances the candidate's utility decreases, as the employer's good impression has of the job candidate decreases. The employer's utility also decreases as the candidate becomes less motivated to work for the company. To facilitate incomplete information there are 3 possible utility structures for each side, which models a long term candidate, short term candidate and compromising candidate. The complete domain including the utility functions is part of the GENIUS framework and available to download from the Internet [2].

4.1 Experiments design

We extended the existing GENIUS negotiation system to include a newly developed chat interface for a WOZ based system using the previously described *KBAgent*. We then studied 32 human participants negotiate interactions with this agent. All participants were students in three different academic institutions, and had different fields of studies. They were highly motivated to attain good scores as they received bonus points to their course grade which is a function of their final utility score in the session.

We then divided these students randomly so that 16 students used the "old" menu based interface, and 16 used the newly developed chat interface. It it important to note that all other parts of the interfaces were **identical**; That is the only visible difference between them was the chat-box instead or pull-down boxes (see Figure 2). The people played the role of the job candidate, while the *KBAgent* played the role of the employer.

Prior to the start of the negotiation task, the people were given a full tutorial about the task at hand, the interface and the possible utility functions. A short test was issued to verify that the subjects understood the instructions and task at hand. The subjects did not know any details regarding the automated agent with which they were matched, or the fact that it was not a human player. The out-

[2]The GENIUS website can be found at http://mmi.tudelft.nl/negotiation/index.php/Genius

come of each negotiation task was either reaching a full agreement, opting out, or reaching the deadline.

In addition, following each session of the experiments (for both interfaces) we conducted a post-experiment questionnaire, in which the subjects had to score on a scale of 1 (lowest) to 5 (highest) the following questions:

- Were the instructions of the experiment clear?
- How happy are you with the negotiation's end result?
- Do you think that your partner was a computer program?
- What do you think was the utility function of your partner? (select 1 out of the 3 options)
- Do you consider the end result to be fair?

4.2 Experimental results

The main goal of the experiments was to check if there are differences in the agent's performance when playing against a human subject who is using a menu-based interface or a chat-based interface.

Table 1: Results of Menu vs. Chat Negotiation Experiments

	Menu-based	Chat-based
Avg. utility - human	398 ($\sigma = 44$)	385 ($\sigma = 41$)
Avg. utility - Agent	**484** ($\sigma = 49$)	**438** ($\sigma = 65$)
Avg. nego. length	5.05	7.25

Table 1 presents the average utility gained by the human players (playing the job candidate) and the *KBAgent* (playing the employer). The standard deviation is written in parenthesis. We can see that the human players got on average similar utility scores, regardless of the interface that they were using. From the agent's perspective, we can see that the agent attained significantly higher scores when faced with partners who were using menu-based interfaces ($p < 0.01$ on a *two-tailed t-test*). Similarly, we can see that the average session length was significantly longer using the chat interface – while the negotiation sessions using the menu interface were on average 5.05 rounds (10 minutes and 6 seconds), the chat interface sessions took on average 7.25 rounds (14 minutes and 30 seconds).

Table 2: Results of Post-Experiment Questionnaire

	Menu-based	Chat-based
instructions	4.21	4.4
happiness	**3.1**	**3.95**
computer program	3.89	3.95
utility type	35%	35%
fairness	**3.26**	**4**

After studying the post-experiment questionnaire, the results of which are summarized in table 2, we can see that both groups understood the instructions very well (4.21 and 4.4 out of 5). Both groups tended to agree that their partner is a computer program (3.89 and 3.95 out of 5), and both groups did not manage to "guess" the type of the utility function of their negotiation partner. That is, only 35% of them (7 students in each group) guessed correctly whether their partner is of a "long term", "short term' or "compromising" type. With respect to their happiness with the results and their grasp of fair outcome, we can see that the chat-based users felt significantly better with the final outcome; They were significantly happier with the end result (3.95 against 3.1, $p < 0.01$ on a *two-tailed t-test*), and felt that the result is fairer (4 against 3.26, $p < 0.05$ on a *two-tailed t-test*).

4.3 Discussion

The above results were somewhat surprising to us as we would have expected exactly the opposite result. That is, we would expected the negotiating agent to attain **lower** utility when playing against a user who is using the menu-based interface. This is because of the following reasons: First, forcing a person to use a preset number of choices in the menu requires her to focus on a limited number of possibilities making the task easier to compute. Second, within the menu-interface, drop-down lists existed for each of the limited choices, allowing the user to see the ordinal relationship of the values inside the list. This allows her to take smaller concession steps and greatly reduces the probability of errors. Last, when selecting the offer from a drop-down lists the utility of the offer is computed and presented automatically to the user, making the task even easier. Thus, we had assumed the person would do better in these case, and consequently, the agent would do worse as the person would achieve higher utility at the expense of the agent.

Thus, our results yielded two key implications: (1) automated negotiators developed for menu-based environment should be somehow **adapted** when migrated to chat-based environments. (2) Humans perceived the outcome of the negotiation session more **positively** when using chat even though their objective utility score remained the same.

Consequently, we focused on the following questions:

> Why does the agent get significantly higher utility when playing against menu-based partners?
>
> How should the next generation of negotiation agents be modified to address this shortcoming?

In addressing these questions, we studied various possible hypothesis to understand these findings and to explain how we should generalize this result. We first present two seemingly obvious explanations which do **not** explain these findings, and then further develop a third hypothesis relating to the nature of people's offers which we believe will need to be addressed in the next generation of negotiation agents.

4.3.1 Rejected Hypotheses

When looking at the causes of the significant difference in utility, a first and intuitive conjuncture is that the discount factor in utility as the time progresses might be a prominent cause. This is an acceptable cause simply because inputing a natural language sentence takes more time than clicking on the dialog boxes. Specifically, Table 2 shows that chat-based sessions takes another 2 rounds on average, which amounts to ≈ -12 utility points.

However, after analytically adding the utility lost due to time discount factor to both groups, the results remains significantly better for when playing against menu players. Specifically, average utility of 524.45 ($\sigma = 32$) against menu players, and 496.65 against chat players ($\sigma = 50$), $p < 0.05$ with the *two-tailed t-test*.

Next, we looked at the time that was spent in the chat experiments due to WOZ translation. This represents the time it took the human behind the system to translate message from natural language to the agent actions model and vice-versa. It appears that on average there were ≈ 274 translation seconds in the chat-based experiments. This amounts to the additional 2 rounds from the previous hypothesis, and correcting them still did not resolve the significant utility advantage when playing against menu-based players.

4.3.2 Accepted Hypothesis - Percentage of Partial Offers

Another interesting observation from the above experiment was that chat-based users sent a higher number of partial agreements than the menu-based users. Specifically, on average the chat-based users sent approximately 2.4 partial offers per session, which amounts to around 40% of their total offers. The menu-based users rarely offered partial agreements even though the interface does not constraint them from doing so, and the instructions explicitly discuss this possibility.

In order to verify this claim we conducted an additional set of experiments in which we did not allow users to send partial offers (unless of course using the specific value of "no agreement" in the minor issues). We did so by issuing a message saying "I prefer to discuss offers with all 6 issues" whenever a partial offer had been sent. Besides that message, we followed exactly the same experiment design as before.

Table 3: Results – Negotiation without Partial Offers

	Menu-based	Chat-based
Avg. utility - human	397 ($\sigma = 39$)	373 ($\sigma = 51$)
Avg. utility - Agent	458 ($\sigma = 82$)	414 ($\sigma = 94$)
Avg. nego. length	6.6	9.4

The experiment was conducted in a similar manner and included 24 participants: 12 played with the menu interface and 12 with the chat interface. The results are depicted in Table 3, and they verify this hypothesis. We found that there is **no** statistical difference between the average utility gained by the agent when playing against these two groups.

When negotiation is conducted using a chat interface, several additional problems arise, such as dialog manager and context resolution. For instance, the following sentence that was sent by a chat user:

```
``I suggest you work 9 hours as a QA."
```

can be interpreted in two ways: a partial offer of the following form {Salary=QA,Hours=9}, or an adaptation of these issues with respect to a previously discussed offer, thus a complete offer with these two new values. Regardless of the interpretation, an automated negotiator that was built around menu-based interface will not have to deal with many partial offers that exist in chat-based negotiation. Therefore, it might be the case that the *KBAgent*'s strategy with respect to partial offers, or specifically its lack of strategy, hindered its performance.

We continued with analyzing the post-experiment questionnaire and now, to our surprise, we did not see any *significant* difference in the groups perception of fairness (3.5 vs. 3.9), or overall happiness with the outcome (3.3 vs. 3.4). The complete data table was omitted due to space constraints.

5. CONCLUSIONS

This paper takes the first step towards moving the problem of automated negotiation towards natural language interfaces. Before tackling the complex problems of NLP and Dialog management, we studied how the current state-of-the-art automated negotiator would perform when paired against chat-based interface. We discovered that the automated negotiation algorithm did not transfer well to more natural forms of conversation. Simply adding a chat-based interface to the existing agent yielded agreements that were significantly *worse* than agreements based on the menu-based interface. In an additional experiment we isolated the reason for the algorithm's inability to cope with partial agreements as the main cause for its decreased performance.

We conclude that future negotiation algorithms for chat environments and other natural interfaces will need to take different strategies from those used by current negotiation agents [14, 3, 10, 17, 19]. While these state of the art agents attempt to find successful agreements on all issues simultaneously, our findings strongly suggest that future agents will instead need to take an issue-by-issue algorithm towards negotiations, or explicitly form partial agreements with people. We are currently studying how this finding can be implemented, and encourage other researchers to do the same.

6. REFERENCES

[1] Mehmet Bac and Horst Raff. Issue-by-issue negotiations: The role of information and time preference. *Games and Economic Behavior*, 13(1):125–134, March 1996.

[2] Lutz-Alexander Busch and Ignatius Horstmann. A comment on issue-by-issue negotiations. *Games and Economic Behavior*, 19(1):144–148, April 1997.

[3] Andrew Byde, Mike Yearworth, Kay-Yut Chen, and Claudio Bartolini. AutONA: A system for automated multiple 1-1 negotiation. In *Proceedings of the 2003 IEEE International Conference on Electronic Commerce (CEC)*, pages 59–67, 2003.

[4] M Keith Chen. Agendas in multi-issue bargaining: When to sweat the small stuff. Technical report, Harvard Department of Economics, Cambridge, November 2002.

[5] Michael H. Coen. Design principles for intelligent environments. In *AAAI/IAAI*, pages 547–554, 1998.

[6] Philip R. Cohen. The role of natural language in a multimodal interface. In *Proceedings of the 5th annual ACM symposium on User interface software and technology*, UIST '92, pages 143–149, New York, NY, USA, 1992. ACM.

[7] Nils Dahlbäck, Arne Jönsson, and Lars Ahrenberg. Wizard of oz studies: why and how. In *Proceedings of the 1st international conference on Intelligent user interfaces*, IUI '93, pages 193–200, New York, NY, USA, 1993. ACM.

[8] Ya'akov Gal, Sarit Kraus, Michele Gelfand, Hilal Khashan, and Elizabeth Salmon. An adaptive agent for negotiating with people in different cultures. *ACM TIST*, 3(1):8, 2011.

[9] P. T. Hoppman. *The Negotiation Process and the Resolution of International Conflicts*. University of South Carolina Press, Columbia, SC, May 1996.

[10] Catholijn M. Jonker, Valentin Robu, and Jan Treur. An agent architecture for multi-attribute negotiation using incomplete preference information. *Autonomous Agents and Multi-Agent Systems*, 15(2):221–252, 2007.

[11] Ron Katz and Sarit Kraus. Efficient agents for cliff edge environments with a large set of decision options. In *Proceedings of the 5th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 697–704, 2006.

[12] J. F. Kelley. An empirical methodology for writing user-friendly natural language computer applications. In

Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '83, pages 193–196, New York, NY, USA, 1983. ACM.

[13] Patrick Kenny, Arno Hartholt, Jonathan Gratch, William Swartout, David Traum, Stacy Marsella, and Diane Piepol. Building interactive virtual humans for training environments. In *Proceedings of Interservice/Industry Training, Simulation and Education Conference (I/ITSEC)*, 2007.

[14] S. Kraus and D. Lehmann. Designing and building a negotiating automated agent. *Computational Intelligence*, 11(1):132–171, 1995.

[15] Raz Lin and Sarit Kraus. Can automated agents proficiently negotiate with humans? *CACM*, 53(1):78–88, January 2010.

[16] Raz Lin, Sarit Kraus, Tim Baarslag, Dmytro Tykhonov, Koen Hindriks, and Catholijn M. Jonker. Genius: An integrated environment for supporting the design of generic automated negotiators. *Computational Intelligence*, to appear.

[17] Raz Lin, Sarit Kraus, Jonathan Wilkenfeld, and James Barry. Negotiating with bounded rational agents in environments with incomplete information using an automated agent. *Artificial Intelligence*, 172(6-7):823–851, 2008.

[18] M. J. Osborne and A. Rubinstein. *A Course In Game Theory*. MIT Press, Cambridge MA, 1994.

[19] Yinon Oshrat, Raz Lin, and Sarit Kraus. Facing the challenge of human-agent negotiations via effective general opponent modeling. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2009.

[20] A. Pease, S. Colton, A. Smaill, and J. Lee. Semantic negotiation: Modelling ambiguity in dialogue. In *Proceedings of Edilog 2002, the 6th Workshop on the semantics and pragmatics of dialogue*, Edinburgh, UK, 2002.

[21] Ben Shneiderman and Catherine Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition)*. Pearson Addison Wesley, 2004.