Constructing utility models from observed negotiation actions

Angelo Restificar
University of Wisconsin-Milwaukee
Milwaukee, WI 53211 USA
angelo@cs.uwm.edu

Peter Haddawy
Asian Institute of Technology
Pathumthani, Thailand 12120
haddawy@ait.ac.th

Abstract

We propose a novel method for constructing utility models by learning from observed negotiation actions. In particular, we show how offers and counter-offers in negotiation can be transformed into gamble questions providing the basis for eliciting utility functions. Results of experiments and evaluation are briefly described.

1 Introduction

Utility functions encode an agent's preference and risk taking behavior, and in many negotiation scenarios they are often considered private information. Although various elicitation techniques for decision makers have been widely used (see for example [Keeney and Raiffa, 1976]), they are not readily applicable in the negotiation scenario. For example, an agent can not ask the the opposing party lottery questions to assess its utility function. The use of learning mechanisms in negotiation has been investigated in several recent studies, see for example [Bui *et al.,* 1999; Zeng and Sycara, 1998], and has been shown to be an effective tool in handling uncertainty and incompleteness. None of the previous studies, however, have directly addressed the issue of utility elicitation. In many cases, value functions and utilities are assumed to be partially or completely known as shown in their experiments and evaluation. In this work, we assume that each party's utility is completely unknown to the other and no trusted third party exists. We describe how information provided by observed negotiation actions can be used to construct the utility model of an agent.

An important tool in eliciting utility functions is the use of lottery or gamble questions and the concept of certainty equivalence. Let D be a domain, U be a utility function over D, and let o_2 and O_j be outcomes in a gamble G where o, occurs with a probability p, O_2 occurs with a probability (1 - p), and $o_i, o_j \in D$. A certainty equivalent is an amount x such that the decision maker (DM) is indifferent between G and $\hat{x}$. Thus, $U(\hat{x}) = pU(o_i) + (1-p)U(o_j)$ or $\hat{x} = U^{-1}[pU(o_i) + (1-p)U(o_j)]$.

2 Negotiation actions as gamble questions

We would like to construct a model that gives us preference and risk-taking information given a position x_i and a gamble that insists on a position X_j. A position is either an offer or a counter-offer. Due to space limitation, our discussion is restricted to constructing the buyer's model. We use Zeuthen's concept of probability to risk a conflict [Zeuthen, 1930] as a basis for transforming negotiation transactions into gamble questions. While Zeuthen assumes complete knowledge of utility functions, our approach attempts to build a utility model by observing negotiation actions.

A conflict occurs when both parties do not reach an agreement. The probability that a party would risk a conflict is proportional to the difference between what the party wants and what it is offered. The closer the other party's offer is to what is desired, the smaller this probability should be. The farther away the other party's offer is from what is desired, the larger the readiness to risk a fight or conflict.

Definition 1 *Let U_B and Us be B 's (buyer) and S 's (seller) utility function, respectively. Let X_B be B 's position and x_s be S's position. The probability that B will risk a conflict, PB. and the probability that S will risk a conflict, ps, are defined as follows:*

$$p_B = \begin{cases} \frac{U_B(x_B) - U_B(x_S)}{U_B(x_B)} & \text{if } U_B(x_B) > U_B(x_S) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$p_S = \begin{cases} \frac{U_S(x_S) - U_S(x_B)}{U_S(x_S)} & \text{if } U_S(x_S) > U_S(x_B) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

By computing the expected utility, B can decide whether to accept S"s offer or insist its own position. In particular, B counter-offers X_B whenever the gamble has a utility higher than *xsJ.e.,* $U_B(x_S) < (1-p_S)U_B(x_B) + p_S U_B$ (conflict). B accepts *xs* if for B the expected utility of the gamble (1 - $p_S)U_B(x_B) + p_S U_B$ (conflict) can not exceed $U_B(x_S)$. If the utility of a non-agreement (conflict) is higher than *xS* then the offer is rejected.

Without losing generality we will assume that U_B is monotonicallv decreasing while *Us* is monotonically increasing. The following theoretical results allow us to generate training instances implied by a counter-offer.

Theorem 1 *Let X_B beB 's counter-offer and x_s be S 's offer Furthermore, let G =* $(1-p_S)U_B(x_B) + p_S U_B$ *(conflict) and let $\hat{x}_B$ be B's certainty equivalent for G. If(G > U_B(XS)) then for any nondegenerate G, $\hat{x}_B \in (x_B, x_S)$.*

Theorem 2 (Inferior Offers) *Let XB be B'S counter-offer andxs be S's offer. Furthermore, letG = (1~PS)UB(ZB) + $p_S U_B$(conflict) and p^*_S be S 's readiness to risk a conflict at x, $x_S < x$. If $(G > U_B(x_S))$ then $((1-p^*_S)U_B(x_B) + p^*_S U_B(\text{conflict})) > U_B(x), \forall\, x$ such that $x_S < x$.*

By Theorem 1, if B decides to make a counter-offer *XB* to an offer x_s by 5, then *B's* certainty equivalent belongs to the interval (x_B, z_S). Theorem 2 states that if B prefers the gamble to an offer of *xs* by S then B would also prefer to gamble than accept any offer by 5 higher than *xs.* Symmetrically, B prefers any offer lower than X_B over a gamble. Counter-offers imply preference between a gamble and a specific offer. Although B's estimate of 5's probability to risk a

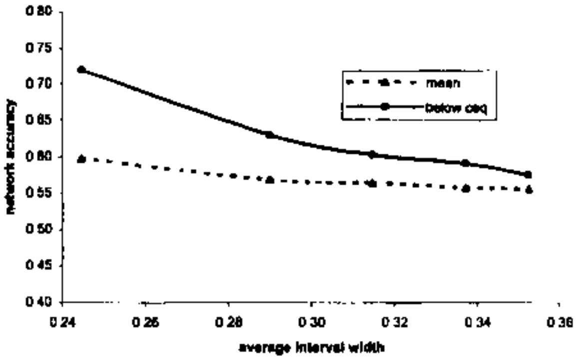

Figure 1: Overall Performance Results

conflict, *ps,* is not specifically known, B's counter-offers are in part based on *ps.* Hence, by learning B's actions, *ps* values are indirectly incorporated into the model.

3 Utility model construction and evaluation

We ran experiments using two control utility functions: $\gamma_1(x) = 1 - \frac{e^{0.05x}}{149}$ (risk-averse, decreasing function) and $\gamma_2(x) = e^{-0.0025x}$ (risk-prone, decreasing function). In each negotiation sequence, *ps* is either generated randomly or is set using {0.50,0.25,0.60} as values. The negotiation using $\gamma_1(x)$ is over the domain $D1$ = [50,100] and that of $\gamma_2(x)$ is over D_2 = [200,700]. The value of conflict c is set at the maximum value of the domain. The buyer-seller negotiation strategy, σ, vary among Boulware-Conceder, Conceder-Boulware, and Conceder-Conceder pairs. We define our Boulware strategy as one where the agent concedes only 10% of the time and a Conceder strategy as one where concession is frequent at 90% of the time. Whenever an agent concedes, concession is randomly chosen between 0—50% of the difference between both agents' most recent counter-offers.

The artificial neural network used in our experiments has one hidden layer with four nodes. The input layer contains three input nodes and the output layer contains two nodes. Data fed into the input layer are scaled so that values only range between 0 and 1. Negotiation sequences used for training, tuning, and testing are randomly generated using a chosen strategy pair, a control utility function $\gamma(x)$, a negotiation domain, and a constant conflict value. We used a k-1 cross validation method to train and tune the network, where k is the number of negotiation transactions (offer-counter pairs) in each negotiation sequence. Network training is stopped when either no improvement in performance is detected for a successive 2,000 epochs or the number of epochs reaches 20,000. Among the data generated using the intervals, 90% is used for training and 10% is used for tuning. We point out that although the generated data maybe learned using other techniques we have chosen to use neural networks for convenience purposes.

We have randomly generated a total of 97 negotiation sequences, wherein each of *B's* counter-offer *B's* expected utility of the gamble is greater than that of S's offer. The total number of negotiation transactions is 477 which gives an average of 5 transactions per negotiation sequence. The training instances are obtained by generating a total of 200 random data points for each observed negotiation transaction; 100 random data points for each of the region below and above the interval. The certainty equivalent, which is obtained from the control utility function, lies inside each interval. For each of the region below and above the certainty equivalent 100 test points are generated. We then evaluate the approach by comparing how well the model performs when trained using the intervals against the test points from the control utility function.

Intuitively, not all negotiation transactions maybe useful. For example, an offer that is near the maximum domain value and a counter-offer that is near the minimum domain value has an interval width that is close to the width of the domain. Since we are using the interval to estimate the certainty equivalent such a negotiation transaction would be less useful to one in which both the offer and counter-offer are closer to the certainty equivalent. Results of regression analysis suggest that useful interval widths are those that are about 50% of the domain width. We have used this result as a basis to eliminate data points that may not be useful in constructing the utility model.

To test the overall performance, negotiation sequences were grouped into subsets where normalized interval widths are no greater than 0.50, 0.45, 0.40, 0.35, and 0.30. The average interval width for each subset is 0.35, 0.34, 0.31, 0.29, and 0.24, respectively. The average network performance of each of these respective subsets is shown in Figure 1. The overall network performance increases as the average interval width corresponding to the negotiation transactions decreases. The solid curve shows the performance of the utility model in predicting whether an offer is preferred by B to a gamble using only implicit data points below the interval. This is important because B's counter-offers only correspond to the lower limit of the interval. The dotted curve shows the accuracy of the model in predicting whether an offer is preferred by B to a gamble and whether B prefers a gamble to an offer. Tne mean accuracy is obtained by averaging the results using implicit data points below *and* above the interval. The results suggest that for intervals with average width of 0.24 the network can predict about 72%: of the time whether an offer is preferred to a gamble by B. For intervals with average width of less than or equal to 0.31, we are able to predict with more than G0% accuracy whether B prefers the offer to a gamble. In addition, the predictive accuracy of the model when implicit data points above and below the interval arc used is better than a random guess. We ran four right-tailed z-tests and one right-tailed t-test using the following hypotheses: $H_0 : \mu = 0.50$ and $H_a : \mu > 0.50$. For the /-test the null hypothesis is rejected at a = 0.005. In each of the r-tests, the null hypothesis is rejected at $\alpha = 0.001$.

In summary, we have outlined theoretical results that would allow us to construct utility models from negotiation actions. Results from our experiments suggest that our utility model provides significant predictive capability.

References

[Bui *et al.*9 1999] H. H. Bui, S. Venkatesh, and D. Kieronska. Learning other agents' preferences in multi-agent negotiation using the bayesian classifier. *International Journal of Cooperative Information Systems,* 8:275-294, 1999.

[Keeney and Raiffa, 1976] Ralph Keeney and Howard Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs.* Cambridge University Press, 1976.

[Zeng and Sycara, 1998] Dajun Zeng and Katia Sycara. Bayesian learning in negotiation. *Int. Journal Human-Computer Studies*, 48:125-141, 1998.

[Zeuthen, 1930] Frederick Zeuthen. *Problems of Monopoly and Economic Warfare.* Routledge and Kegan Paul, Ltd., 1930.