

A Scalable Method for Online Learning of Non-linear Preferences Based on Anonymous Negotiation Data

D.J.A. Somefun

CWI, Center for Mathematics and Computer
Science
P.O. Box 94079, 1090 GB Amsterdam, The
Netherlands
koye@cw.nl

J.A. La Poutré

CWI and Technical Univ. Eindhoven
School of Techn. Management
P.O. Box 513, 5600 MB Eindhoven, The
Netherlands
hlp@cw.nl

ABSTRACT

We consider the problem of a shop agent negotiating *bilaterally* with *many* customers about a bundle of goods or services together with a price. To facilitate the shop agent's search for mutually beneficial alternative bundles, we develop a method for online learning customers' preferences, while respecting their privacy. By introducing extra parameters, we represent customers' highly nonlinear preferences as a linear model. We develop a method for learning the underlying stochastic process of these parameters online.

Categories and Subject Descriptors

I.2.6 [Artificial Intelligence]: Learning; I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search

General Terms

Economics, Algorithms

Keywords

Negotiation, Market-Based Methods, Online Learning

1. INTRODUCTION

We consider the problem of a shop agent negotiating *bilaterally* with *many* customers about a bundle (configuration)—an array of bits, representing the presence or absence of each of the shop's goods in the bundle—together with a price for the bundle. Like the work of [3, 2, 4, 1], the techniques developed in this paper try to benefit from the so-called win-win opportunities that arise with multi-issue negotiations.

The methods proposed in Faratin et al. [2], and Coehoorn and Jennings [1] are geared towards finding win-win opportunities through modeling the preferences of the negotiation partner, for issues with independent valuations. This paper follows a similar approach; like Klein et al. [3] and our

earlier work [4], we consider *interdependencies* between issues, however, which make the problem considerably harder. The novelty of our current approach lies in the introduction of a new scalable and very fast method for *online* learning these complex customer preferences, while respecting their privacy. To respect customers' privacy, the method as developed only uses *anonymous* data on past negotiations: i.e., the shop cannot determine if he has dealt with a customer in the past. As the conducted computer experiments show, the necessary aggregate information, to significantly facilitate the search for mutually beneficial alternative bundles in future bilateral negotiations, is learned quickly for relatively large problems.

2. NEGOTIATION PROCESS

The shop sells a total of n goods, each of which may be either absent or present in a bundle, so that there are $2^n - 1$ distinct bundle-configurations containing at least 1 good, with $n = 20$. A negotiation concerns a bundle, together with a price for that bundle, and it is conducted in an alternating exchange of offers and counter offers. An example of such a practice may involve the sales of bundles of news items in categories like politics, finance, economy, sports, arts, etc.

During the negotiation, the shop will suggest a *limited number* of alternative bundles. Hopefully, these suggestions will lead to the bargainers finding mutually beneficial alternative bundles. Such bundles represents *Pareto improvements* because switching makes at least one bargainer better off without making the other worse off. Ideally, the shop will eventually find a *Pareto efficient* bundle: a bundle for which no Pareto improvements exists.

Typically, a customer initiates the negotiation process by submitting an initial bundle and opening price. The shop agent constructs an ordered set from all the bundles that lie in the initial bundle's neighborhood (i.e., all bundles which in binary representation have a hamming distance of 1 to the initial bundle); it orders the bundles based on the likelihood that they will result in Pareto improvements, starting with the most likely bundle. By advising the first bundle in this set first, the second bundle second, and so on, the shop agent searches (locally) for Pareto improvements (cf. [4] for further details). To assess the likelihood of a Pareto improvement requires statistical knowledge of customers' preferences. The focus of this paper lies on how we can obtain this knowledge based on anonymous negotiation data.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.
Copyright 2006 ACM 1-59593-303-4/06/0005 ...\$5.00.

3. PROBLEM STATEMENT

First we show how we can rewrite customers' highly non-linear preferences as a linear model. In the absence of strategic behavior and stochasticity in customers' preferences, this will give use a direct solution to the learning problem. In the next section, we use the obtained insights to develop a fast method for online learning more about customer preferences given strategic behavior and stochasticity.

There are n individual goods from which customers can compile the bundle they wish to purchase. The column vector $\mathbf{b} = [b(1), \dots, b(n)]^T$ denotes the binary representation of a bundle (i.e., $b(i) = 1$ if and only if good i is part of the bundle). Let a_i denote a customer's valuation for obtaining good i individually and $a_{i,j}$, $a_{i,j,l}$, etc. denote the synergy effects exclusively caused by obtaining a subset of goods $\{i, j\}$, $\{i, j, l\}$, etc.: e.g., $a_{i,j}$ is equal to the customer's valuation for bundle $\{i, j\}$ minus the valuation for obtaining the goods individually. There are $\binom{n}{1}$ parameters of the type a_i , $\binom{n}{2}$ parameters of the type $a_{i,j}$, and so on; thus there are at most $k = \sum_{i=1}^n \binom{n}{i} = 2^n - 1$ parameters. In the experiments, we consider problems where $n = 20$. To make the number of parameters manageable we assume that beyond subsets of size 3 the synergy effect are always zero: i.e., there are only $k = \binom{n}{1} + \binom{n}{2} + \binom{n}{3}$ number of parameters.

Given some (arbitrary but) fixed encoding, denoted by $\langle \cdot \rangle$, we can map all k parameters into the $(k\text{-dimensional})$ vector $\mathbf{a}$. We can now define a customer's monetary valuation for bundle $\mathbf{b}$, $v_{\mathbf{a}}^c(\mathbf{b})$, as follows:

$$v_{\mathbf{a}}^c(\mathbf{b}) = \sum_{i=1}^n \mathbf{a}_{\langle i \rangle} b(i) + \sum_{i=1}^n \sum_{j=i+1}^n \mathbf{a}_{\langle i,j \rangle} b(i)b(j) + \dots + \mathbf{a}_{\langle 1,\dots,n \rangle} b(1)b(2)\dots b(n). \quad (1)$$

From Eq. (1) (and encoding $\langle \cdot \rangle$), we can conclude that the seller can construct a mapping $\mathbf{f} : \{0, 1\}^n \mapsto \{0, 1\}^k$. For all customers, their monetary valuation for any bundle $\mathbf{b}$ can be expressed as a linear function of their k -parameters and $\mathbf{f}(\mathbf{b})$. For example, if there are only 3 goods—good 1, 2, and 3—then $\mathbf{b} = (1, 1, 0)$ could (depending on the encoding) give $\mathbf{f}(\mathbf{b}) = (1, 1, 0, 1, 0, 0, 0)$. Irrespective of the encoding, $\mathbf{f}(1, 1, 0)$ should contain three 1's and four 0's: i.e., the parameters $a_{\langle 1 \rangle}$, $a_{\langle 2 \rangle}$, and $a_{\langle 1,2 \rangle}$ contribute to the valuation of bundle $(1, 1, 0)$.

In the absence of strategic behavior and stochasticity, $\mathbf{a}$ is the same for all customers. The seller can then just obtain $\mathbf{a}$ by solving a system of linear equations. For example, suppose there are t observations and for each $i = 1, \dots, t$ the seller has the observations v_i and $\mathbf{b}_i$, where v_i denotes customers' valuations for bundle $\mathbf{b}_i$. Then

$$\mathbf{v} = \mathbf{B}\mathbf{a} \quad (2)$$

with $\mathbf{v} = [v_1, \dots, v_t]^T$ and $\mathbf{B} = [\mathbf{f}(\mathbf{b}_1), \dots, \mathbf{f}(\mathbf{b}_t)]^T$ (thus $\mathbf{B}$ is a $t \times k$ matrix). Since the seller also knows $\mathbf{f}$, he knows both $\mathbf{v}$ and $\mathbf{B}$. Now if the rank of matrix $\mathbf{B}$ is equal to k , then the (Moore-Penrose) generalized matrix inverse of $\mathbf{B}$, denoted by $\mathbf{B}^+$, can be used to compute $\mathbf{a}$: i.e., $\mathbf{a} = \mathbf{B}^+ \mathbf{v}$.

Typically, customer preferences differ from one customer to the next: i.e., $\mathbf{a}$, which completely describes customers' preferences, is a random vector. We assume $\mathbf{a}$ is normally distributed with a vector of means $\mu_{\mathbf{a}}$ and covariance matrix $\Sigma_{\mathbf{a}}$. Moreover, since customers will behave strategically the seller will not directly observe their valuations.

4. LEARNING METHOD

The method we develop for estimating the vector of means $\mu_{\mathbf{a}}$ and covariance matrix $\Sigma_{\mathbf{a}}$, contains (a) a sub method for learning $\mu_{\mathbf{a}}$ and $\Sigma_{\mathbf{a}}$ in the absence of strategic behavior and (b) a sub method for preprocessing the negotiation data. First method (b) is applied to the data such that a sufficient amount of the noise caused by strategic behavior is removed.

(Sub Method A) As was discussed in Section 3, per customer we have an instance of the random vector $\mathbf{a}$, which completely specifies her preferences. For a customer c , let $\mathbf{a}_c$ denote such an instance. Suppose the seller would negotiate for a very long time with this customer c , while constantly suggesting different bundles. In the absence of strategic behavior, eventually the seller would have collected the negotiation data $\mathbf{v}_c = [v_{c1}, \dots, v_{ct}]^T$, and $\mathbf{B}_c = [\mathbf{f}(\mathbf{b}_{c1}), \dots, \mathbf{f}(\mathbf{b}_{ct})]^T$ such that $\mathbf{B}_c$ has rank k . Analogous to Eq. (2), the seller can then just obtain $\mathbf{a}_c$ by solving $\mathbf{v}_c = \mathbf{B}_c \mathbf{a}_c$: i.e., $\mathbf{a}_c = \mathbf{B}_c^+ \mathbf{v}_c$, where $\mathbf{B}_c^+$ is again the generalized matrix inverse of $\mathbf{B}_c$.

For large k , bargainers will break off the negotiations or reach a deal, before the matrix $\mathbf{B}_c$ grows into a rank k matrix. Therefore, the seller suggests instead a sequence of bundles such that the union of the goods contained in these bundles has strictly less goods than is possible. (In the experiments, 4, 5, or at most 6 individual goods are being considered.) In that case, relative to k the matrix $\mathbf{B}_c$ contains only a few nonzero columns. Whenever k' , the rank of the matrix $\mathbf{B}_c$, equals the number of nonzero columns, we can obtain $\mathbf{a}'_c$ —the k' -dimensional vector containing all the identifiable parameters of $\mathbf{a}_c$ —by solving $\mathbf{v}_c = \mathbf{B}'_c \mathbf{a}'_c$ (i.e., $\mathbf{a}'_c = \mathbf{B}'_c^+ \mathbf{v}_c$); we get the matrix $\mathbf{B}'_c$ by removing all zero columns from $\mathbf{B}_c$. Based on these newly computed parameter values, it is now relatively straightforward to obtain estimations of the mean, and covariance matrix.

(Sub Method b) With strategic behavior the seller does not observe $\mathbf{v}_c = [v_{c1}, \dots, v_{ct}]^T$. Instead he observes $\mathbf{p}_c = [p_{c1}, \dots, p_{ct}]$, the sequence of bids submitted by the customers, where there is generally a difference between p_{ci} and v_{ci} . To deal with strategic behavior requires some implicit or explicit model of customers' bidding strategies. For example, in [4, 1] they learn more about customer preferences based on the difference between two consecutive bids. This implies some implicit model of a customer's bidding behavior, however. We want to make this model explicit so that we can check whether the model is consistent with a customer's bidding behavior.

Strategic behavior of customers can roughly be split into a fixed and variable component. The fixed component specifies how much a customer c is at most willing to bid for a bundle $\mathbf{b}$ worth $\mathbf{f}(\mathbf{b})^T \mathbf{a}_c$, irrespective of the current state of the negotiation; the variable component specifies how a customer concedes through time.

Filtering out the fixed component of a customer's bargaining strategy is impossible. Especially in the case of bundling, the seller can eradicate most of the second type of noise, however. The idea is that at the end of a negotiation process the seller will use some of the gathered negotiation data to model how the opponent concedes through time; the rest of the data can then be used to check whether the developed model correctly predicts how a customer concedes through time. It is very unlikely that such a test is passed incorrectly (error of the first kind) because both the correctness of the estimated values for the identifiable parameters ($\mathbf{a}'_c$) and the concession model are tested jointly. The other pos-

sibility, of not being able to model a negotiation process is even less problematic; the seller can just simply ignore this negotiation. In real world applications the seller could have a collection of typical models and check which of the models fits the data of a particular negotiation best. Consequently, enough negotiation data can be preprocessed and fed to sub method (a) to ensure good estimations of μ_a and Σ_a .

5. NUMERICAL EXPERIMENTS

Customers are at most willing to pay $x\%$ of their true valuation; for each customer x is randomly drawn between 85 and 95. Moreover, customers and shop use time dependent bidding strategies: the offer submitted by a customer (shop) is monotonically increasing (decreasing) in both the number of bidding rounds and the valuation.

We compare our new approach of having *no a priori information* and learning customer preferences online (as discussed in Section 4), to the one where—for example because of expert knowledge—the shop already knows the underlying joint probability distribution of all bundle valuations. Besides comparing our new online procedure (referred to as *ML*) with the previous method (called *S*), we also perform the same series of experiments with a benchmark procedure (called *B*), which simply recommends a random bundle from the current bundle's neighborhood. That is, the benchmark does not base the order in which it recommends the next bundle on the estimated likelihood of Pareto improvements, as the *ML* and *S*-systems do.

Customers initiate the negotiation by submitting an offer containing an initial bundle b_{init} . To give the shop some room for improvement, we initialize the customer's initial bundle by randomly selecting a bundle b which, in binary representation, has a Hamming distance of 5 to the bundle b^* that is associated with the highest gains from trade.

Both *ML* and *S*-system greatly outperform the *B*-system. To illustrate how well the *ML*-system performs compared to the *S*-system, figure 1 shows the relative performance and the number of rounds within which a deal is reached (if a deal is reached in the first place) per customer. Relative performance specifies the percentage of the maximal obtainable Pareto improvements (given the starting point) that is actually realized at the end of a negotiation. To simplify the comparison we hard wired the *ML*-system to stop exploring after 500 customers. (Remember, learning is completely online, however.) In the exploitation phase of the *ML*-system, the two systems generate virtually the same performance. Figure 1 illustrates the great advantage of using negotiation data. In the exploration phase 500 customers suffices for the *ML*-system, because per customer the system obtains roughly 55 high quality data points.

6. CONCLUSIONS

We conduct computer experiments with simulated customers that have highly *nonlinear* preferences and are interested in bundles containing up to 20 individual goods. We compare our new method of having *no a priori information* and learning about customer preferences online, to the one where—for example because of expert knowledge—the shop already knows the underlying joint probability distribution of all bundle valuations. Our experiments show how, after 500 customers, the two approaches have virtually the same performance. Both methods significantly increase the

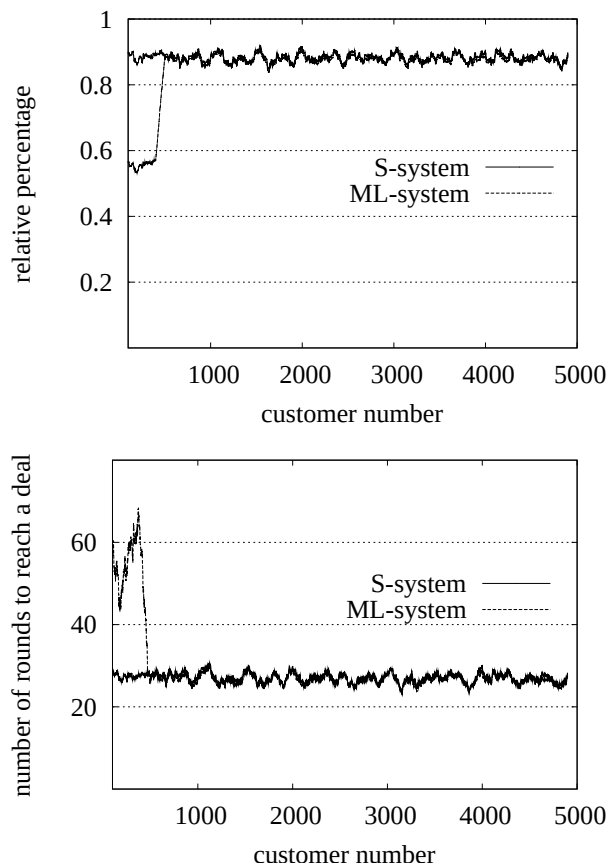


Figure 1: 100-step moving average of the relative % and the number of rounds to reach a deal.

speed with which deals are reached, as well as the number and the Pareto efficiency of the deals reached, compared to a benchmark. Moreover, the experiments show that the new procedure is able to learn the necessary information online, even if due to strategic behavior customers' best offers are significantly less than their reservation values.

7. ACKNOWLEDGMENTS

This work builds on earlier joined work with Tomas Klos.

8. REFERENCES

- [1] R. M. Coehoorn and N. R. Jennings. Learning an opponent's preferences to make effective multi-issue negotiation tradeoffs. In *Proc. 6th Int Conf. on E-Commerce*, pages 59–68. ACM press, 2004.
- [2] P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make issue trade-offs. *Journal of Artificial Intelligence*, 142(2):205–237, 2003.
- [3] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. *Group Decision and Negotiation*, 12:111–125, 2003.
- [4] D. J. A. Somefun, T. B. Klos, and J. A. La Poutré. Online learning of aggregate knowledge about nonlinear preferences applied to negotiating prices and bundles. In *Proc. 6th Int Conf on E-Commerce*, pages 361–370. ACM press, 2004.