An arbitration agent for bilateral trading in an electronic marketplace

K. Balaji, Varun S. Malhotra
IBM India Research Lab
Block 1, Indian Institute of Technology,
Hauz Khas, New Delhi - 110 016.
{kbalaji,varunmal}@in.ibm.com

Abstract

In the context of business to business (B2B) commerce, bilateral trading on products or services would involve multiple attributes. The negotiating agents in the marketplace have conflicting interests and always act selfishly to maximize their own gains. An arbiter agent by knowing the interest of these agents could resolve the conflicts between them. That is the arbiter agent could ensure fairness among these agents by recommending bids that would make none worse off than the other. This objective of the arbiter agent, is modelled as a max-min optimization problem and we prove that this problem is *NP-hard*. We propose a heuristic based on *simulated annealing* to solve this optimization problem and validate our claims that this heuristic behaves well in practice by an elaborate set of simulations.

Index Terms

Bilateral trading, agent based negotiation, arbitration, bid recommendation, algorithmic hardness, and simulated annealing.

I. Introduction

TRADING mechanisms among businesses could be broadly classified as, (i) negotiation - in which a buyer (or supplier) initiates the negotiation on a product or service and the suppliers (or buyers) would negotiate with the buyer (or supplier) till it ends up in a deal, (ii) auctions - in which the supplier puts a product in auction and multiple buyers bid on it till a deal is made, (iii) reverse auctions - in which buyer announces the auction and suppliers submit their bids till the buyer chooses the supplier(s) from whom the goods would be procured (or an agreement is signed in the case of services), (iv) double auctions - in which buyers and suppliers post their bids and offers in a marketplace (a trading exchange) and transactions happen whenever they overlap. In business to business (B2B) trading, exchange of goods or services, by any of the trading mechanisms, involve multiple attributes such as quantity, price, delivery time and so on. Therefore, in such exchanges the trading parties would have conflicting interests, for example, the buyers always trying to procure goods in large quantities within short period of time for low prices and suppliers committing to deliver only a partial amount of the quantity requested within the specified period and for higher prices. Hence, the role of an arbitrator becomes of paramount importance in such scenarios for resolving the conflicts that would arise in such trades. Arbitrator based trading had been studied in [2], [7], [8], [11], [14], [15], [16], [17], [18] and [19]. Exchange mechanisms which are reliable and fair, to match the trading parties and mediate on their conflicting interests had been studied in [18] and [19].

To ensure fairness in a two-party trading, in this paper, we propose a max-min optimization model to recommend Pareto optimal bids (bids that would make none of the participating agents worse off than the other). As the max-min optimization problem is algorithmically hard, we propose a heuristic to solve this problem. The proposed heuristic is based on simulated annealing, which was introduced in [10]. Simulated annealing is a heuristic technique that has attracted significant attention for solving algorithmically hard optimization problems[1][9][13]. The method of simulated annealing is an analogy with thermodynamics, specifically with the way that liquids freeze and crystallize, or metals cool and anneal. The essence of the process lies on how the liquid or metal under consideration is cooled. In the same way, the simulated annealing algorithm depends very much on the cooling schedule[4]. Considering algorithms for minimization problems, rapid cooling or quenching (that is by taking rapid downhill moves in the search space) may greedily converge to a local minimum, but not necessarily a global minimum. Traversing the search space reasonably slower by exploring the uphill moves too, would converge to a global minimum quickly.

The rest of the paper is organized as follows. In Section II, finding a possible deal between a buyer and a supplier which would minimize their conflicts and would be of high utility to both of them, in a two-party multiple issue arbitration, is modelled as a max-min optimization problem. This optimization problem is proved to be NP-hard, in Section III. In Section IV, we propose a heuristic based on *simulated annealing* to solve this problem. We validate our claim that this heuristic is applicable in practice through an elaborate set of experiments in Section V. Finally, we conclude in Section VI.

II. Problem definition and Modelling

Multi-attribute utility theory architecture for two-party negotiation without involving any mediator had been studied in [3]. In this work, we study two-party multi-attribute negotiation in the presence of an intermediary. The buyer and the supplier agents express their utilities on all the attributes (and for different values of these attributes) to the arbiter (the mediator). The objective of the arbiter is to propose a deal which is fair - balance the gains of both the parties, and of high utility to both of them. The arbitrator′s objective is modelled as an optimization problem, in this section.

In this paper, we restrict the problem to the utility functions of the following type. Let $A_1,\ A_2,\ A_3,\ldots,\ A_k$ be the non-discrete attributes and $A_{k+1},\ A_{k+2},\ldots,\ A_n$ be the discrete attributes, on which the users specify their utilities. Let the attribute values for the i^{th} non-discrete attribute be in one of these disjoint intervals $I_{i_1}, I_{i_2}, \ldots, I_{i_{l(i)}}$ (that is the attribute value specified on A_i is a number in one of these disjoint intervals). It is assumed that the buyer and the supplier would specify constant utilities over each interval I_{i_j}, where $1 \leq j \leq l(i)$ and i is the index corresponding to the non-discrete attribute A_i.

The problem of proposing a bid which is fair and of high utility to the trading parties in a two-party multi-issue trade could be modelled as the following optimization problem, which is referred as the Max-Min Utility Optimization Problem (MMUOP).

`MMUOP` : Let $S = S_1$ X S_2 X...X S_k, that is the Cartesian product of the sets $S_1, S_2, \ldots, S_k$, where $S_i = \{I_{i_1}, I_{i_2}, \ldots, I_{i_{l(i)}}\}$ if the attribute corresponding to the index i is a non-discrete attribute and S_i is the collection of all possible values of A_i if it is a discrete attribute. Let the buyer and the supplier specify utilities for all $x_i \in S_i$, where $1 \leq i \leq k$. The objective is $max_{x \in S} min\{f(x), g(x)\}$, where the total utility functions $f(x)$ and $g(x)$ are the sum (or weighted sum) of the utilities specified on $x_i \in S_i$, if $x = (x_1, x_2, \ldots, x_k)$. The objective function of the above model is motivated by the fairness function (which is also called as Rawlsian function) used in social economics [12]. Intuitively, the model MMUOP maximizes the total utility of the worst off individual, among the buyer and the supplier, evaluated at each point of the bargaining region. The optimum solution of MMUOP is a Pareto optimum solution, as it dominates all other solutions over the region.[1] In the model, the motivation for considering the fairness function, maximising the minimum utility of the worst off individual, as compared to other fairness functions such as, maximising the sum of utilities or minimising average utility of both the individuals, is that the former has the *egalitarian*[2] property, whereas, the latter has the *utilitarian*[3] property [12].

A solution x (which is also a feasible solution) of MMUOP is a k-tuple, $(x_1, x_2, \ldots, x_k)$ where $x_i \in S_i$. A neighbor of a solution $x = (x_1, x_2, \ldots, x_k)$ of MMUOP is another solution $y = (y_1, y_2, \ldots, y_k)$ of MMUOP, where $x_i = y_i$, except for an r where $1 \leq r \leq k$ (refer to figure 1). For example, consider a product or service with two attributes A_1 and A_2. Let the attributes A_1 and A_2 take attribute values v_1, v_2, v_3 and w_1, w_2, w_3, respectively. That is, $S = S_1 \mathrm{X} S_2$, where $S_1 = \{v_1, v_2, v_3\}$ and $S_2 = \{w_1, w_2, w_3\}$. For the solution (v_1, w_1), its neighbors are (v_1, w_2), (v_1, w_3), (v_2, w_1) and (v_3, w_1).

III. Algorithmic complexity of the problem

In this section, we prove that the problem of obtaining a solution which would be of high utility to both the buyer and the supplier using the max-min metric is *NP-hard*.

---

[1] A social optimal economic outcome in a competitive marketplace is called a Pareto optimal outcome if it is impossible to make some players better off without making some other individuals worse off [12]. The same definition applies here as the MMUOP model is a social economic model for a competitive market.

[2] A social welfare function is egalitarian if its aim is to distribute the gains equally between the trading parties.

[3] A social welfare function is utilitarian if increase or decrease in individual utilities translate into identical changes in social utility. So, the function remains neutral towards making none worse off than the other.

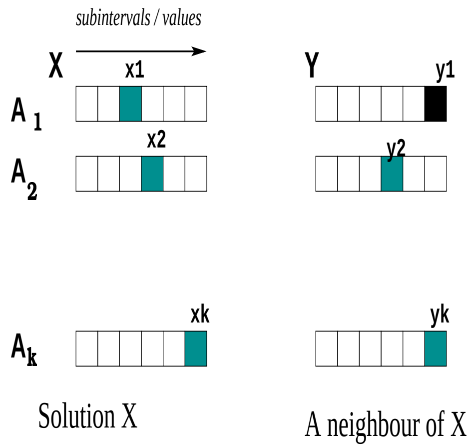

Fig. 1. An example showing a neighbor of a solution X

`Theorem :` MMUOP is *NP-hard*.

`Proof :` Weighted-Set-Partition (WSP) problem is known to be NP-Complete [6]. To prove the theorem, we reduce an instance of WSP to an instance of MMUOP. The WSP problem is the following.

Instance I : A set $S = \{s_1, s_2, \ldots, s_n\}$ and weights w_i attached with each s_i for all $1 \leq i \leq n$.

Question Q : Is there a partition of S into sets A and $S - A$ such that $\sum_{s_i \in A} w_i = \sum_{s_i \in S-A} w_i$?

For an instance I of WSP, consider the following instance I' of MMUOP.

Instance I′ : For an instance I of WSP, let the indices i of s_i correspond to the indices i of A_i, where A_i s are the attributes over which the buyer and the supplier specify their utilities. Let the utility values be specified over the intervals I_{i_1} and I_{i_2}, if the i^{th} attribute is a continuous attribute (that is the buyer and the supplier specify constant utilities over the intervals I_{i_1} and I_{i_2}) and the utility values be specified over the attribute values v_{i_1} and v_{i_2}, otherwise. The specified utility values of I_{i_1} and I_{i_2} (similarly, the utility values of v_{i_1} and v_{i_2}) be either 0 or w_i where w_i is the weight of s_i for the instance I of WSP. Furthermore, let the specified utilities be such that if the buyer has set w_i then the supplier has set 0 and vice versa.

Output : An element x of S such that x is an optimal solution, that is $min\{f(x), g(x)\} =$

$max_{x \in S} min\{f(x), g(x)\}$, where $S = S_1 X S_2 X \ldots X S_n$ such that $S_i = \{I_{i_1}, I_{i_2}\}$, if i is an index corresponding to a continuous attribute and $S_i = \{v_{i_1}, v_{i_2}\}$, otherwise.

We note that the question Q for an instance I of WSP can be answered yes (or no), if and only if, for the instance I' of MMUOP which corresponds to the instance I of WSP, the output x satisfies (or does not satisfy) $f(x) = g(x)$ and vice versa. This proves that MMUOP is NP-hard. ■

The following theorem states that there is no non-exhaustive deterministic algorithm to solve MMUOP efficiently.

`Theorem :` The running time complexity of every non-exhausitive deterministic algorithm for MMUOP is as worse as that of the exhaustive algorithm.

`Proof :` In [5], it has been proved that the running time of every non-exhaustive algorithm to solve the following problem in database theory is approximately equal to the running time of its exhaustive algorithm.

`Database Problem :` A database object has grades associated with each of its m attributes. The problem is to find the top k objects whose overall score (which is obtained by combining the attribute grades using fixed monotone aggregation function[4] or combining rule, such as min or average)is higher than the other objects in the database.

It can easily seen that MMUOP is reducible to the above problem as its objective function is monotonic. ■

Based on the above theorem, in our simulation study, we compared the probabilistic algorithm of MMUOP proposed in this paper only with the exhausitive algorithm.

[4]An aggregation function f is monotonic if $f(x_1, x_2, \ldots, x_m) \leq f(x'_1, x'_2, \ldots, x'_m)$ whenever $x_i \leq x'_i$ for all i.

IV. HEURISTIC

The problem of obtaining a bid which would maximize the minimum utility of both the buyer and the supplier when both of them specify utilities over individual attributes is proved to be an NP-hard problem in the previous section. In this section, we propose a heuristic based on *simulated annealing*. The term *simulated annealing* derives from a roughly analogous physical process of heating and then, slowly cooling a substance to obtain a strong crystalline structure. In *simulated annealing*, a minimum of the cost function (the function that is considered for optimization) corresponds to a ground state of the substance. The simulated annealing process lowers the temperature by slow stages until the system *freezes* and no further changes occur. At each temperature, the process must proceed long enough for the system to reach a steady state or equilibrium. This is known as *thermalization*. The time required for thermalization is the *De-correlation time*; correlated micro-states are eliminated. The sequence of temperatures and the number of iterations applied to thermalize the system at each temperature comprise an annealing schedule. To apply simulated annealing, the system is initialized with a particular configuration. A new configuration is constructed by imposing a random displacement. If the energy of this new state is lower than that of the previous one, the change is accepted unconditionally and the system is updated. If the energy is greater, the new configuration is accepted probabilistically. This is the fundamental procedure of *simulated annealing*. The fundamental procedure allows the system to move consistently towards lower energy states, yet still *come* out of local minima due to the probabilistic acceptance of some upward moves. If the temperature is decreased logarithmically, simulated annealing guarantees a better solution than in the case when the temperature is decreased exponentially. On the other hand, the system converges to a solution much faster when the temperature schedule is exponential than when it is logarithmic. Hence, there is a trade-off between the time of convergence to a solution and the optimality of the solution.

The following heuristic outputs an optimal (or a nearly optimal) solution to the optimization problem `MMUOP`.

The above heuristic proceeds as follows. First, an initial solution $InitSoln$ is chosen randomly from the

---

Heuristic :

1) Initialize the temperature, T.
2) Select the initial solution, $InitSoln$, at random.
3) Set current solution, $CurrSoln = InitSoln$ and $BestSoln = InitSoln$.
4) Repeat steps $5-15$ until $TerminateHeuristic =$ $true$.
5) Set $TempSoln$ as a random neighbor of CurrSoln.
6) If utility value of $TempSoln >$ utility value of $CurrSoln$ then goto step 7 else goto step 10.
7) Set $CurrSoln = TempSoln$.
8) If utility value of $CurrSoln >$ utility value of $BestSoln$ then goto step 9 else goto step 14.
9) Set $BestSoln = CurrSoln$ and goto step 14.
10) Generate a random number $RandNum$, such that $0 \leq RandNum \leq 1$.
11) Set $Diff =$ utility value of $TempSoln-$ utility value of $CurrSoln$.
12) If $RandNum < e^{\frac{Diff}{T}}$ then goto step 13 else goto step 14.
13) Set $CurrSoln = TempSoln$.
14) Reset the temperature, T.
15) Set $TerminateHeuristic = True$, if terminating condition is reached.

---

set of all solutions of MMUOP. The temperature T, is initialized to a constant value, for the first iteration. Depending on the cooling schedule chosen, for the subsequent iterations T is defined as follows,

- linear : $T(i) = \frac{b}{i}$,
- exponential : $T(i) = b * a^i$,

- logarithmic : $T(i) = \frac{b}{ln(i+a)}$,

where a, b are constants, and i is the iteration index. At every iteration, CurrSoln is the solution to be examined in that iteration and BestSoln is the best solution encountered till that iteration. In an iteration, the heuristic randomly chooses a neighbor of CurrSoln, which is referred as TempSoln, and moves deterministically or probabilistically to TempSoln depending on whether the move improves or worsens the objective. Finally, the heuristic terminates when it reaches a terminating condition, such as the number of iterations has exceeded a certain threshold, or the running time set for the algorithm is reached, etc.

`Recommending multiple bids :` Multiple Pareto (nearly-Pareto) optimal bids can be obtained by running the heuristic multiple times. These Pareto optimal bids could be recommended by the arbiter agent to the trading agents for further negotiation.

V. Experimental Results

A. Simulation study

We experimented the heuristic for solving `MMUOP`, proposed in Section IV, on various cooling schedules - linear, exponential and logarithmic. Furthermore, the experiments were done for various utility distributions - random, decay, normal and skewed-normal. In our experiments, for each attribute A_k, we fixed the number of subintervals for which the individual utilities are specified, to be 10. So, if there are m attributes the size of the solution space would be 10^m.

B. Empirical Benchmarking

In this section, we present the results obtained after experiments on various utility distributions. For each utility distribution the behaviour of the heuristic with respect to the cooling schedules - linear, exponential and logarithmic, are compared. In the case of utility distributions - decay, normal and skewed-normal, for each attribute A_k, the utility value for the sub-interval indexed α, is assigned according to the function $cf_k(\alpha)$, where $\sum_{\alpha=1}^{10} cf_k(\alpha) = 1$, and $cf_k(\alpha)$ is a decay, normal or skewed-normal distribution, as the case may be.

The heuristic was run on 50 samples with respect to each utility distribution. On each sample the heuristic was run 100 times. For each run, the simulated annealing steps were iterated till it reached the global optimum (that is the solution obtained by exhaustive search). The time ratio (Exhaustive Search/Simulated Annealing) in all the plots is averaged over these runs. In all the plots, as the time ratio is plotted in exponential scale, we note that the behaviour of the proposed heuristic for small number of attributes is not visible. However, for these cases, we observed in our experiments, that the heuristic converged to the global optimum in much lesser time than the exhaustive search.

Random distribution :

For every attribute A_k, the 10 different sub-intervals are assigned random utilities between 0 and 1, for both the parties (the buyer and the supplier). The results of the experiments are shown in Figure 2 and Figure 3. The experiments show that as the number of attributes increase the time ratio of exhaustive search versus simulated annealing increases rapidly. It can be seen from Figure 3 that the time ratio (Time taken by Exhaustive search/Time taken by Simulated Annealing) increases exponentially (linearly in the log scale). Figure 2 shows that the proposed heuristic is very much ahead, in terms of time, compared to exhaustive search for more than 6 attributes. In this case, all the cooling schedules perform almost equally. Among them, logarithmic schedule performs better than the other two.

Decay distribution ($f_k(\alpha) = e^{-\alpha}$) :

The utility values for one party (buyer or supplier) is assigned as follows. For every attribute A_k, the utility values for the sub-intervals are assigned according to the function $f_k(\alpha) = e^{-\alpha}$, where $1 \leq \alpha \leq 10$. The fact that the buyer and the supplier would have complementary utilities imply that the utility values of the other party are assigned according to the function $1 - f_k(\alpha)$. The obtained results are shown in Fig. 4 and Fig. 5. The exponential cooling-schedule performs far better than the other two cooling schedules in this

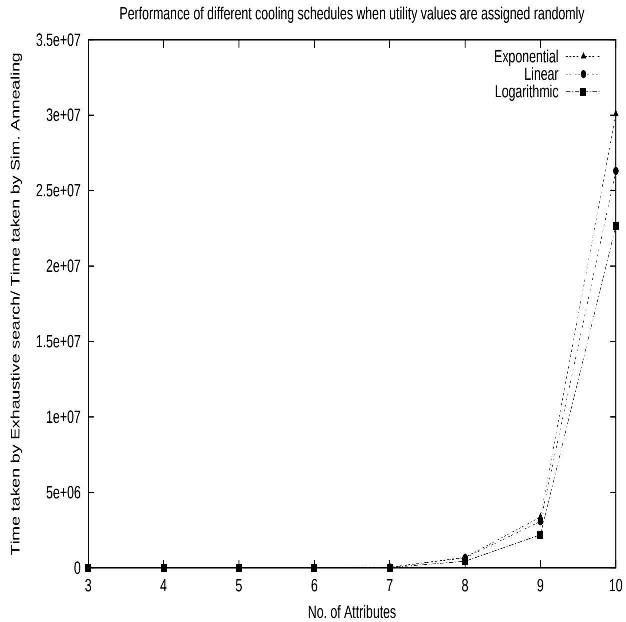

Fig. 2. Different cooling schedules are compared when utilities are assigned randomly

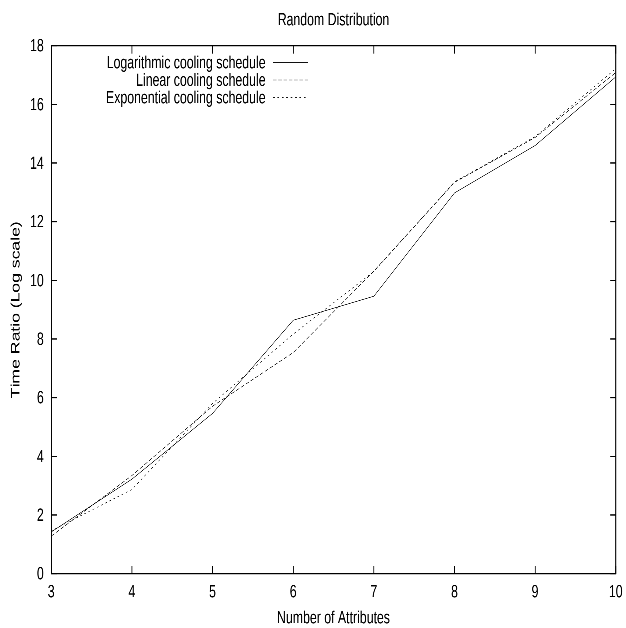

Fig. 3. The performance of various cooling schedules in logscale for random distribution

case (we refer to Fig. 4). The exponential increase of the time ratio with the number of attributes is shown in Fig. 5.

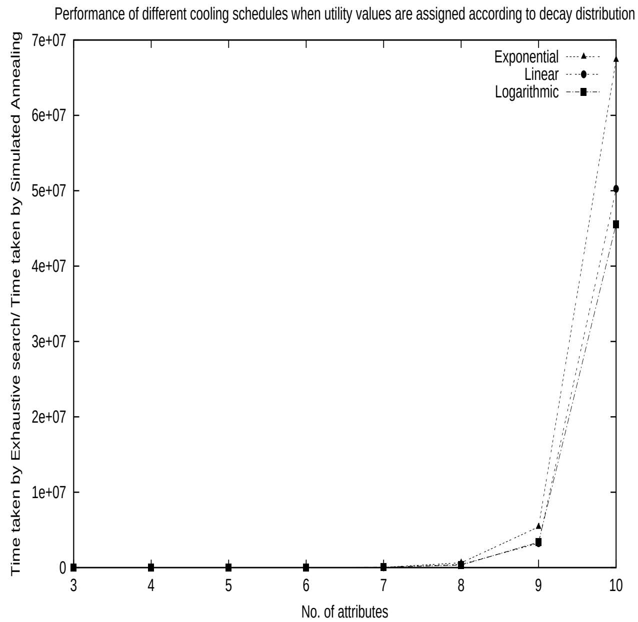

Fig. 4. Different cooling schedules are compared when utilities are assigned using decay distribution

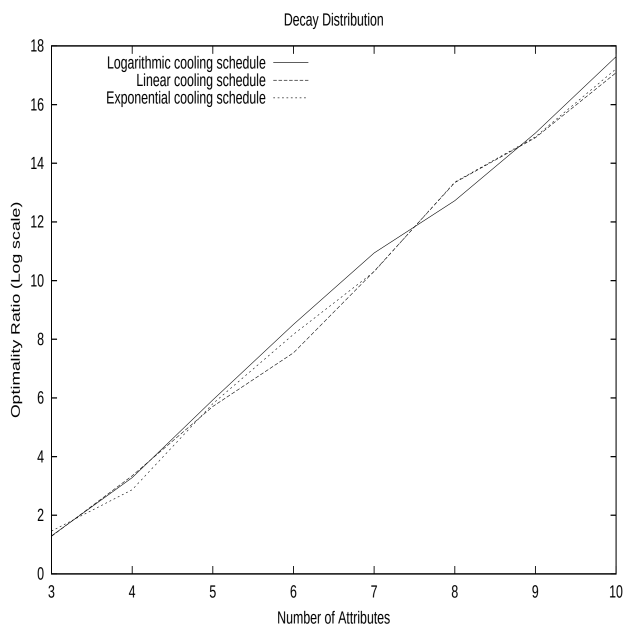

Fig. 5. The performance of various cooling schedules in logscale for decay distribution

Normal distribution ($f_k(\alpha) = e^{-(\alpha-5)^2}$) :

As in the case of decay distribution, for one of the parties (buyer or supplier), utilities are assigned using a normal distribution, and the other is assigned utilities based on its complementary distribution. That is, for the buyer (or the supplier), the utility values for the sub-intervals of each attribute A_k are assigned according to the function $f_k(\alpha) = e^{-(\alpha-5)^2}$, where $1 \leq \alpha \leq 10$. For the other party it is assigned according to the function $1 - f_k(\alpha)$. Assigning utilities based on the normal distribution with mean 5 would imply at interval 5 the utility is high and utilities decrease symmetrically for the intervals to the left and right of interval 5. The results of this case are shown in Figure 6 and Figure 7. As shown by these results, the exponential cooling schedule is the best in terms of time of convergence (we refer to Fig. 6). However, it is interesting to note that all the three cooling schedules are not very far apart in terms of their performance. Figure 7 shows the exponential increase of time ratio with the number of attributes, for this case.

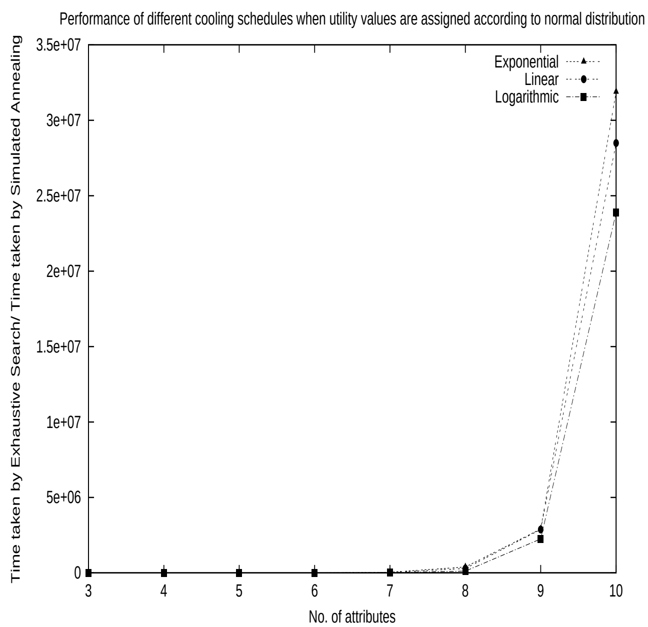

Fig. 6. Different cooling schedules are compared when utilities are assigned using normal distribution

Skewed-normal distribution ($f_k(\alpha) = log(\alpha)e^{-(\alpha-5)^2}$):

Similar to the last two cases, one of the parties (buyer or supplier) is assigned utilities using a skewed-normal distribution, and the other its complement. More precisely, for the buyer (or the supplier), the utility values for the sub-intervals of each attribute A_k are assigned using the function $f_k(\alpha) = log(\alpha)e^{-(\alpha-5)^2}$, where $1 \leq \alpha \leq 10$. The other party is assigned utilities using the function $1 - f_k(\alpha)$. Even though, the highest utility value is assigned at interval 5 in both the normal and the skewed-normal case, both of them differ in

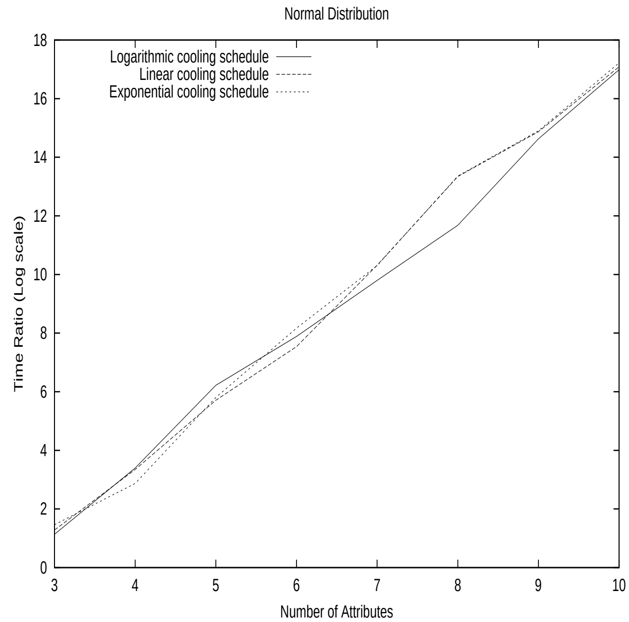

Fig. 7. The performance of various cooling schedules in logscale for normal distribution

terms of symmetry. That is in the skewed-normal case too, utilities decrease for the intervals to the left and right of interval 5, but unsymmetrically. As is evident from the results (we refer to 8), the linear cooling schedule performs nearly as good as the exponential cooling schedule. The logarithmic schedule is very much behind the exponential and linear. But, on the whole all these cooling-schedules perform badly on the solution space generated in this case compared to the solution space generated in other cases, that is, random, decay and normal. For this case, the exponential increase of time ratio as the number of attributes increase is shown by Figure 9.

C. Discussion

The crucial point to note in all our simulation results is that the proposed heuristic, which is based on simulated annealing, gave similar results as the exhaustive search in much lesser time. The results hold even if the number of intervals with respect to each attribute increase. In general, the time ratio, exhaustive search/simulated annealing, grows exponentially as the number of attributes or the number of ranges increase. Therefore, in negotiations which involve many attributes or attributes with multiple ranges, the proposed heuristic is highly recommended.

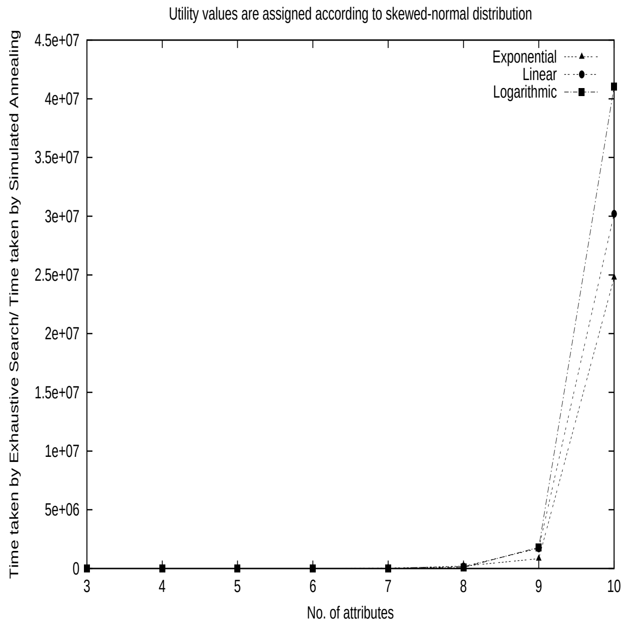

Fig. 8. Different cooling schedules are compared when utilities are assigned using skewed-normal distribution

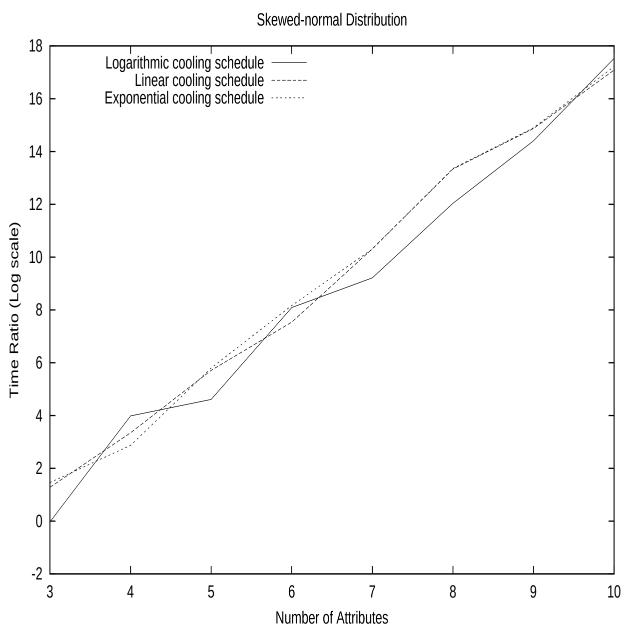

Fig. 9. The performance of various cooling schedules in logscale for skewed-normal distribution

It is quite intuitive to expect that the behaviour of the heuristic is not only dependent on the cooling schedule chosen, but also on the nature of the solution space which is being explored. For instance, if the solution space being explored is very smooth and has very few local optima, any fast converging cooling schedule would converge to an optimal solution very fast. On the other hand, if the solution space has many number of local optima then a fast converging cooling schedule will take time to converge to a global optimum, since it would freeze quickly to these local optima and would incur more time to get out of them.

Comparing the cooling schedules - linear, exponential and logarithm, on which we have experimented in this paper, the exponential cooling schedule is the best in terms of time of convergence to the global optimal solution when there are very few local optima. However, when the solution space has many local optima, logarithm or linear schedule would behave better or atleast as good as the exponential schedule. Looking closely at the experiments exactly this is what happens. In the case of decay distribution, the exponential schedule outperforms the other two schedules, because there are very few local optima in the solution space. Whereas, in the case of random, normal and skewed-normal atleast one of the cooling schedules, linear or logarithm performs closely to the exponential schedule, because there are multiple local optima in the solution space for these cases. For the random distribution case, all the cooling schedules behave equally well, because as a consequence of choosing the utilities randomly, multiple hills and valleys are getting formed in the solution space.

VI. Conclusion

Electronic agents trading in a marketplace would have conflicting interests and always act selfishly to maximize their own gains. To resolve such conflicts among the trading agents, a trusted intermediary could interfere and propose bids that are agreeable to all of them. In this paper, we have modelled this problem as a Max-Min Utility Optimization Problem (MMUOP) for bilateral trading. Recommending a bid which would be agreeable and of high utility to both the trading agents had been proved to be algorithmically hard in Section III. Therefore, we propose an optimization heuristic based on the method of *simulated annealing* to solve this problem. Furthermore, we validate that the proposed heuristic behaves well in practice with an elaborate set of experiments. In our experiments, we tested the heuristic on various utility distributions with different cooling schedules. The heuristic is observed to behave well in all these test cases. The simulation results reveal that the heuristic give the similar results as the exhaustive search in much lesser time.

ACKNOWLEGEMENT

The authors would like to thank Ravi Kothari for his valuable suggestions in the preparation of this paper.

REFERENCES

[1] C.R.Aragon, D.S.Johnson, L.A.McGeoch and C.Schevon, *Optimization by simulated annealing: An experimental evaluation*, Part II, graph coloring and number partitioning, Operations Research (39), 1991.

[2] J.Bailey and Y.Bakos, *An exploratory study of the emerging role of electronic intermediaries*, International Journal of Electronic Commerce, 1(3), 1997.

[3] Mihai Barbuceanu and Wai-Kau Lo, *A multi-attribute utility theoretic negotiation architecture for electronic commerce*, Proceedings of Agents 2000, Barcelona, Spain, pp. 239-246.

[4] Harry Cohn and Mark Fielding, *Simulated Annealing - Search for an optimal temperature schedule*, SIOPT, Volume 9, Number 3, 1999.

[5] Ronald Fagin, Amnon Lotem and Moni Naor, *Optimal Aggregation Algorithms for Middleware*, PODS, 2001.

[6] Michael R.Garey and David S.Johnson, *Computers And Intractability: A Guide to the Theory of NP-Completeness*, Freeman, SanFrancisco, 1979.

[7] R.H.Guttman and P.Maes, *Agent-mediated integrative negotiation for retail electronic commerce*, In Proc. of the Workshop on Agent Mediated Electronic Trading, Minneapolis, Minnesota, May 1998.

[8] R.H.Guttman, A.Moukas, and P.Maes, *Agent-mediated electronic commerce : A survey*, Knowledge Engineering Review, June 1998.

[9] D.S.Johnson, C.R.Aragon, L.A.McGeoch and C.Schevon, *Optimization by simulated annealing: An experimental evaluation*, Part I, graph-partitioning, Operations Research (37), 1989.

[10] S.Kirkpatrick, C.D.Gelatt.Jr. and M.P.Vecchi, *Optimization by simulated annealing*, Science 220(4598), 1984.

[11] R.D.Luce, and H.Raiffa, *Games and Decisions*, Dover Publications Inc., Mineola, New York, 1989. Reprint. Originally published : New York, John Wiley and Sons, 1957.

[12] A.Mas-Colell, M.D.Whinston, and J.R.Green, *Microeconomic Theory*, Oxford University Press, 1995

[13] R.H.J.M.Otten and L.P.P.P.van Ginneken, *The Annealing Algorithm*, Kluwer, Boston, 1989.

[14] H.Raiffa, *The Art and Science of Negotiation*, Harvard University Press, Cambridge, Mass., 1982.

[15] P.Resnick, R.Zeckhauser and C.Avery, *Roles for electronic brokers*, Tech. Report 179, MIT Center for Coordination Science, 1994.

[16] Tuomas Sandholm, *eMediator : A next generation electronic commerce server*, AAAI Workshop Technical Report WS-99-01, pp. 46-55.

[17] M.B.Sarkar, B.Butler and C.Steinfield, *Intermediaries and cybermediaries : A continuing role for mediating players in the electronic marketplace*, Journal of Computer Mediated Communication, 1(3), 1995.

[18] Thomas Tesch, Peter Fankhauser and Aris M. Ouksel, *Arbitration Protocols for Competing Software Agents*, IJCIS 10(1-2), 2001.

[19] Thomas Tesch and Peter Fankhauser, *Arbitration and Matchmaking for Agents with Conflicting Interests*, CIA 1999.