DOI: 10.2298/CSIS120306054T

Agent Negotiation on Resources with Nonlinear Utility Functions

Xiangrong Tong and Wei Zhang

School of Computer Science, Yantai University
Shandong 264005, China
txr@ytu.edu.cn

Abstract. To date, researches on agent multi-issue negotiation are mostly based on linear utility functions. However, the relationship between utilities and resources is usually saturated nonlinear. To this end, we expand linear utility functions to nonlinear cases according to the law of diminishing marginal utility. Furthermore, we propose a negotiation model on multiple divisible resources with two phases to realize Pareto optimal results. The computational complexity of the proposed algorithm is polynomial order. Experimental results show that the optimized efficiency of the proposed algorithm is distinctly higher than prior work.

Keywords: Nonlinear utility function, Multi-agent Systems, Multi-issue Negotiation, Resources Allocation, Incomplete Information.

1. Introduction

Allocation of multiple divisible resources under incomplete information is always an interesting problem where negotiation is one of essential approaches, such as the work of Luo et al [1], Saha et al [2], Chevaleyre et al [3], Fatima et al [5]. They proposed Pareto optimal negotiation, envy-free negotiation, and welfare optimal negotiation in their researches. In our past work [6–8], we also studied some problems of multi-issue negotiation.

However, prior researches mostly did not take explicit utility functions, or directly regarded resources as agents' utilities. Therefore, only linear utility functions were considered. However, there are much cases where utility functions are nonlinear. For example, $100 will produce lots of utilities if an agent has not a bean, while $100 will produce few utilities if that agent has a capital of a hundred million dollars. The utilities of autonomous agents are some kinds of subjective belief, and resources are objective things, i.e. resources generally have firing effect on utilities.

So the relationship between utilities and resources is usually saturated nonlinear just as indicated by Wooldridge [9]. Furthermore, the relationship should follow the law of diminishing marginal utility according to microeconomics, i.e. the increments of utilities decrease along with the increments of resources. Recently, some researches gradually paid more attention to nonlinear utility functions such as [11, 12, 21–24].

We try to expand linear utility functions to nonlinear cases in this paper. This work is motivated by the sigmoid function of artificial neuron of artificial neural network. In particular, we propose an improved nonlinear utility function to describe the relationship between utilities and resources. The proposed function not only follows the law of diminishing marginal utility, but also is consistent with the assumption of Wooldridge et al.

Sequentially, we bring forward a negotiation model on multiple divisible resources. In particular, we firstly take resources as indivisible units and reach a preliminary agreement using strict alternation in the first phase. Information about the preferences and firing rate of the opponent is also obtained at the same time. Subsequently, we divide resources using a greedy algorithm in the second phase and realize Pareto optimal results.

The first contribution of this paper is the proposal of a new utility function which follows the law of diminishing marginal utility while it is mostly ignored by previous work. Moreover, in the first phase of negotiation, strict alternation provides not only a preliminary agreement but also some useful information for agents. Therefore, history information is not necessarily required. Secondly, this model can generate Pareto optimal results for all negotiation agents. The computational complexity of the proposed algorithm is polynomial order and it is usually lower than that of Fatima et al. Experimental results show that the optimized efficiency of the proposed algorithm has an advantage over the work of Fatima et al. The application of this work includes many similar domains such as allocation of band width, allocation of computational resources, and allocation of business etc.

The rest of this paper is organized as follows. Section 2 describes the basic notions about negotiation scenes and utility functions. We propose a negotiation model and a feasible optimal algorithm in section 3. Section 4 discusses negotiation with complete information, and section 5 discusses negotiation under incomplete information. Experimental settings and results are provided in section 6. Background and related work is provided in section 7. Conclusions are drawn in the last section.

2. Basic Notions

In this section, we describe some basic notions about negotiation such as negotiation scenes, information states of agents, a nonlinear utility function, marginal utility function, and the ratio of marginal utility function. We also present some useful properties of the proposed nonlinear utility function.

2.1. Negotiation Scenes

We improve the negotiation scenes of Fatima et al [5] where agent a and b negotiate over m ($m \geq 2$) divisible resources (issues). We assume that the m resources are independent each other. We let agent a firstly make a bid for instance, and agent b is taken as the opponent.

Definition 1. *Negotiation scenes*

$$N = \langle a, b, R, m, n, \rho, t\rangle \tag{1}$$

where, N denotes a negotiation scene. a, b denote agents who are participating the negotiation. $R = \{r_1, r_2, \ldots, r_m\}$ denotes divisible resources to be allocated. they also can be regarded as issues of negotiation. m denotes the number of resources to be allocated. n denotes the deadline of negotiation, namely the maximum turn of negotiation is n. $\rho = \{\rho_1, \rho_2, \ldots, \rho_m\}$ denotes discount factors of resources ($0 \leq \rho_i \leq 1$). $t \in \mathcal{N}$ denotes the current turn of negotiation. $\mathcal{N}$ denotes natural number.

Definition 2. *Information state of agents*

$$I_a = \langle k_a, x_a, C_b, P_b\rangle \tag{2}$$

Where, I_a is the information state of agent a. $k_a = \{k_a^1, k_a^2, \ldots, k_a^m\}$ denotes resources/issues preferences of agent a. In which, k_a^i denotes the preference of agent a for r_i. We assume that $k_a^i \in [0, 1], \sum_{i=1}^{m} k_a^i = 1$. In this paper, we assume that the preference of agent is invariable along with the quantity of resources allocated. It is a fixed factor for the agent initially. $x_a = \{x_a^1, x_a^2, \ldots, x_a^m\}$ denotes the proportion of resources allocated to agent a. In which, $x_a^i \in [0, 1]$ denotes the proportion of resource r_i allocated to agent a. $C_b = \{C_b^1, C_b^2, \ldots, C_b^l\}$ denotes l possible types of agent b known by agent a in advance. The type of agent b is known by agent a if negotiation is in complete information environment. $P_b = \{P_b^1, P_b^2, \ldots, P_b^l\}$, in which, P_b^i denotes the probability of case that agent b is type C_b^i.

2.2. A Nonlinear Utility Function

There are various researches on nonlinear utility functions of negotiation recently. Some of researches focused on general nonlinear utility function and did not give practical nonlinear utility function, some of researches took Bell function such as [12]. Most of researches did not provide any rules to select nonlinear utility functions synthetically. To this end, we present these rules illustrated as follows.

Most models in economics and finances assume a non-decreasing utility function with diminishing marginal utility. While Arrow-Pratt's definition of absolute risk aversion is related to concave utility functions which implies decreasing marginal utility.

Rule 1. *The utility function should be separable, cumulative, and continuous. Such a function should satisfy the demand of negotiation on multiple divisible resources.*

Rule 2. *The utility function should be bounded. That is to say that the utility of agent would not increase infinitely.*

Rule 3. *The utility function should be monotonously nondecreasing. It means that agent hope to get more resources, no matter how much he has obtained.*

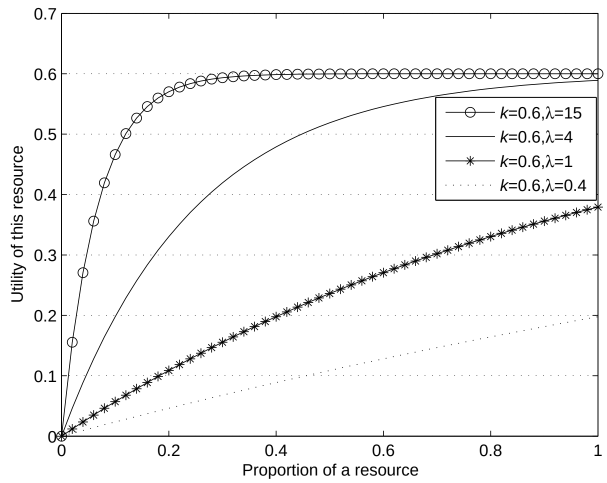

Fig. 1. Relationship between utilities and resources

Rule 4. *The utility function should be a concave function. It means that the increment of utility is monotonously non-increasing.*

Rule 5. *The utility function should follow the property of Arrow-Pratt absolute risk aversion. When an agent has more of one resource, he would confront more risk about this resource.*

Due to the above five rules of utility function, we are motivated by the sigmoid function of artificial neuron. The relationship between utilities and resources(money) indicated by Woodridge is absolutely suitable for sigmoid function. So we propose an improved nonlinear utility function through revising sigmoid function illustrated in Fig. 1.

Definition 3. *An improved nonlinear utility function*

$$u_a^i = k_a^i(1 - e^{-\lambda_a x_a^i}) \tag{3}$$

where, u_a^i denotes the nonlinear utility of agent a for resource r_i. $\lambda_a \in (0, \inf)$ denotes the firing rate of resources to the utilities of agent a. It shows the impact factor of resource to utility. For simplicity, we assume that λ_a is same to all resources of agent a.

The more λ_a is, the faster u_a^i increases. Especially, u_a^i is a step function when $\lambda_a \to \inf$. The utility of agent a can not obtain k_a^i even if all of r_i are allocated to it when $\lambda_a < 4$ because agent a can not be satisfied by the existing resource.

The proposed utility function is different from the prior functions. In particular, when $\lambda_a \leq 1$, the utility function is just approximately linear. Therefor, the prior work of Fatima et al is a particular one of our nonlinear utility function when λ is small. So our proposed utility function is more general and common, including linear and nonlinear cases concurrently.

Definition 4. *The total utilities of agents*

$$U_a = \sum_{i=1}^{m} u_a^i (\rho_i)^{t-1} \tag{4}$$

U_a denotes the total utilities of agent a for all resources. t denotes the current turn of negotiation. u_a^i denotes the utilities of agent a for resource r_i which is defined in definition 3.

Definition 5. *Marginal utility function of agents in nonlinear domain*

$$\frac{du_a^i}{dx_a^i} = \frac{d(k_a^i(1 - e^{-\lambda_a x_a^i}))}{dx_a^i} = k_a^i \lambda_a e^{-\lambda_a x_a^i} \tag{5}$$

It is shown that the marginal utility in nonlinear domain is related to not only the preferences of agents, i.e. k_a^i, but also the firing rate of resources to the utilities of agent a, i.e. λ_a, as well as the quantity of resources allocated to agent a, i.e. x_a^i.

If we take linear utility function, $u_a^i = k_a^i x_a^i$, then the marginal utility is as follows.

Definition 6. *Marginal utility function of agents in linear domain*

$$\frac{du_a^i}{dx_a^i} = \frac{d(k_a^i x_a^i)}{dx_a^i} = k_a^i \tag{6}$$

So the marginal utility of linear function is k_a^i.

Definition 7. *The ratio of marginal utility function*

$$\theta_i = \frac{(u_b^i)'}{(u_a^i)'} \tag{7}$$

where, θ_i denotes the ratio of the marginal utility function of agent b to that of agent a.

If we use a greedy algorithm to generate counter-offer, agent a wishes to maximize its own utilities after satisfying the requirement of agent b's utilities. It divides the m pies such that it reserves the maximum possible shares for those resources where θ_i is low, and gives to agent b the maximum possible shares for those resources where θ_i is high. Thus, agent a would begin by giving agent b the maximum possible shares for the resource with the highest θ_i. It then does the same for the resource with the next highest θ_i and repeat this process until utility requirement of agent b is satisfied. Therefore, the remainders of resources are allocated to agent a to maximize agent a's utilities. In this way, not only the requirement of agent b's utilities is quickly realized, but also the maximum utilities of agent a is obtained.

Assumption 1. *If the proportion of r_i allocated to agent a is x_a^i, then the proportion of r_i allocated to agent b is ($1 - x_a^i$). The higher the value of θ_i is, the earlier the resource r_i is allocated to agent b.*

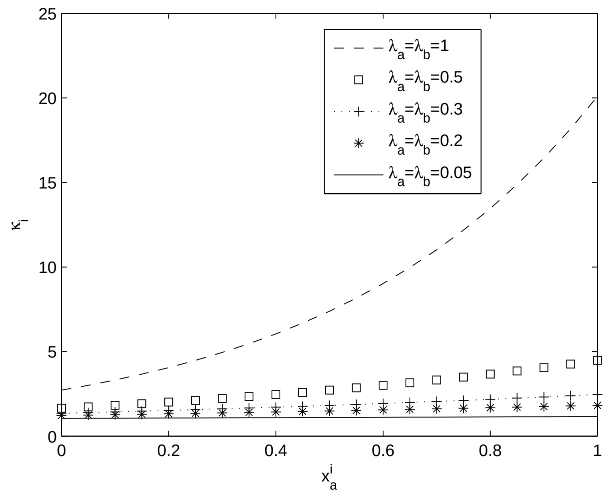

Fig. 2. The relationship among θ_i and x_a^i, λ_a, λ_b

$$\theta_i = \frac{k_b^i \lambda_b e^{-\lambda_b x_b^i}}{k_a^i \lambda_a e^{-\lambda_a x_a^i}} = \frac{k_b^i \lambda_b e^{-\lambda_b (1-x_a^i)}}{k_a^i \lambda_a e^{-\lambda_a x_a^i}} = \frac{k_b^i \lambda_b e^{-\lambda_b}}{k_a^i \lambda_a} e^{(\lambda_a+\lambda_b)x_a^i} \tag{8}$$

According to greedy algorithm, agent a should always choose the highest θ_i at point x_a^i, whenever k_b^i/k_a^i is. Particularly, the choice of resource is depended on x_a^i, as well as $k_a^i, k_b^i, \lambda_a, \lambda_b$. In a word, the only criteria is θ_i. **End.**

Theorem 1. *The value of θ_i of nonlinear utility function is related to not only k_a^i, k_b^i, but also λ_a, λ_b and the quantity of resources allocated to agent a, i.e. x_a^i.*

Proof: see formula (8). **End.**

The relationship among the value of θ_i and x_a^i, λ_a, λ_b is illustrated in Fig. 2. When $\lambda_a = \lambda_b \leq 0.3$, the value of θ_i is approximately invariable, no matter how much x_a^i is.

Theorem 2. *The value of θ_i of linear utility function is $\frac{k_b^i}{k_a^i}$.*

Proof: Due to
$\frac{du_a^i}{dx_a^i} = \frac{d(k_a^i x_a^i)}{dx_a^i} = k_a^i$, $\theta_i = \frac{(u_b^i)'}{(u_a^i)'} = \frac{k_b^i}{k_a^i}$. **End.**

We can find that the work of Fatima where $\frac{k_b^i}{k_a^i}$ is adopted is just the rank of θ_i of linear utility function.

2.3. Properties of the Proposed Nonlinear Utility Function

The proposed utility function follows the following properties which are important for micro-economics. The utility function is bounded, is a concave function, follows the law of diminishing marginal utility, and displays increasing absolute risk aversion according to Arrow-Pratt [10].

Property 1. $u_a^i \in [0, k_a^i)$ *is bounded.*

Proof: Known $\lambda_a > 0, x_a^i \geq 0$, u_a^i is minimal when $x_a^i = 0$, $u_a^i = k_a^i(1 - e^0) = 0$, while u_a^i is maximal when $x_a^i = 1$, $u_a^i = k_a^i(1 - e^{-\inf}) \to k_a^i$ if $\lambda_a \to \inf$. **End**.

Property 2. u_a^i *follows the law of diminishing marginal utility.*

Proof: Known $\lambda_a > 0, x_a^i \geq 0$, $(u_a^i)' = k_a^i \lambda_a e^{-\lambda_a x_a^i} \geq 0$. It shows that u_a^i is monotonously nondecreasing. Particularly, u_a^i is monotonously increasing if $k_a^i > 0 \wedge \lambda_a > 0$.

$(u_a^i)'' = (k_a^i \lambda_a e^{-\lambda_a x_a^i})' = -k_a^i(\lambda_a)^2(1 - e^{-\lambda_a x_a^i}) \leq 0$ is monotonously non-increasing. Particularly, u_a^i is monotonously decreasing if $k_a^i > 0 \wedge \lambda_a > 0$. **End**.

Property 3. u_a^i *is a concave function.*

Proof: Known $\lambda_a > 0, x_a^i \geq 0$, $(u_a^i)' = k_a^i \lambda_a e^{-\lambda_a x_a^i} \geq 0$. $(u_a^i)'' = (k_a^i \lambda_a e^{-\lambda_a x_a^i})' = -k_a^i(\lambda_a)^2(1 - e^{-\lambda_a x_a^i}) \leq 0$, so u_a^i is a concave function. **End**.

Property 4. u_a^i *displays increasing absolute risk aversion.*

Proof: The Arrow-Pratt measure is an attribute of a utility function. If we denote a utility function by $u(x)$. The Arrow-Pratt measure of absolute risk aversion is defined by:

$\gamma(x) = -u''(x)/u'(x)$.

So $\gamma(x_a^i) = \frac{k_a^i(\lambda_a)^2(1-e^{-\lambda_a x_a^i})}{k_a^i \lambda_a e^{-\lambda_a x_a^i}} = \lambda_a(e^{\lambda_a x_a^i} - 1)$. We know that $\gamma(x_a^i)$ also increases when x_a^i increases. It means that the risk aversion of agent a increases when x_a^i increases. Therefore, u_a^i displays increasing absolute risk aversion. Therefore, when agent a possesses lots of this resource, he prefers to reject rather than accept it. **End**.

3. Negotiation Model

In this section, we first discuss the proposed greedy algorithm for generating optimal proposals. Sequentially, we discuss the conditions of acceptance for offers.

3.1. Generating Optimal Proposals

Fatima et al [5] used a greedy algorithm to realize a Pareto optimal result. Just like knapsack problem, the greedy algorithm ranks k_b^i/k_a^i. Actually, agent a indeed ranks θ_i which is linear because $(u_a^i)' = (k_a^i x_a^i)' = k_a^i$. However, the utility function in this paper is nonlinear so that it may be difficult to choose resources.

We propose an improved greedy algorithm to obtain Pareto optimal solutions with nonlinear utility functions.

$$\begin{gathered} max \sum_{i=1}^m u_a^i \\ s.t. \sum_{i=1}^m u_b^i = U_b^I \end{gathered} \tag{9}$$

where, U_b^I denotes the sum of agent b's utilities that must be satisfied when agent a uses a greedy algorithm.

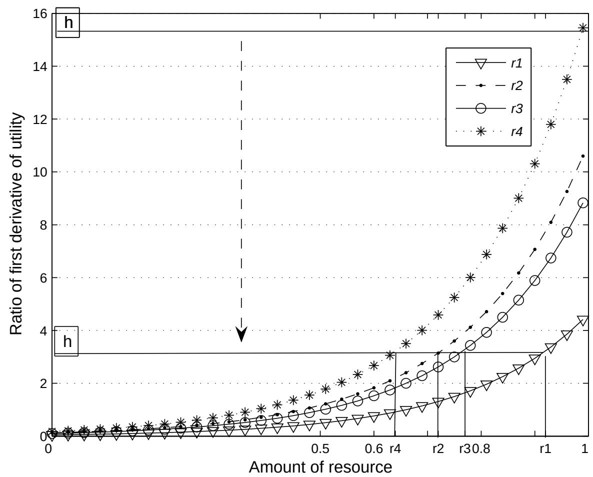

Fig. 3. The relationship between θ_i and x_a^i

According to the above explanation, the choice of resource is depended on θ_i. So we should always choose the biggest θ_i and try to satisfy the requirement of agent b's utilities. Because the value of θ_i is depended on x_a^i, so the biggest θ_i will change among different resources.

Example 1. For example, if $m = 2, n = 2, k_a^1 = 0.4, k_a^2 = 0.6, k_b^1 = 0.5, k_b^2 = 0.5, \lambda_a = 2, \lambda_b = 3$, then when $x_a^1 = 0.7, x_a^2 = 0.4$, $\theta_1 = 3.09 > \theta_2 = 0.46$, when $x_a^1 = 0.4, x_a^2 = 0.8, \theta_1 = 0.69 < \theta_2 = 3.40$. So we find that the biggest θ_i changes from resource 1 to resource 2 when the quantity of resources changes. More examples are illustrated in table 1.

Table 1. The change of the biggest θ_i along with x_a^i

x_a^1	x_a^2	θ_1	θ_2	The biggest θ_i
0.7	0.4	3.09	0.46	θ_1
0.4	0.8	0.69	3.4	θ_2
0.4	0.2	0.69	0.17	θ_1

The proposed approach draws horizontal line in Fig. 3 from the top to the bottom. It means that the value of θ_i decreases from the biggest one until agent b is satisfied. The value of abscissa of the point of intersection with the line of θ_i is x_a^i, and $(1 - x_a^i)$ of the resource is allocated to agent b. Agent a occupies all the remainders of this resource, as long as agent b is satisfied. According to the greedy algorithm, agent a can obtain its Pareto optimal results.

The process of optimization is illustrated in algorithm 1, in which, ω is the amount of pieces of each resource divided by two agents, χ is the initial value of θ_i, ι is the step size of θ_i decreasing, u_b^I is the initial utility of agent b, δ is a toleration of θ_i's error. $u_b^p(i)$ is the optimal utilities of agent b, while $u_a^p(i)$ is the optimal utilities of agent a, $r_a(i)$ is the the share of resource i allocated to agent a.

Algorithm 1

Set $x = 0 : 0.01 : 1; s = (\lambda^b/\lambda^a)(k^b/k^a)e^{-b}$;
$u_b^p(i) := 0, (i = 1, ..., m)$;
while $\Sigma_{i=1}^m u_b^p(i) < u_b^I$ //satisfying agent b's utility
  for $i = 1 : m$ //m resources
    $j := \omega + 1$ //each resource is divided into ω pieces
    while $abs(\theta_i(j) - \chi) > \delta$
      $j := j - 1$;
    end while
    $u_a^p(i) := u_a^i(j); u_b^p(i) := u_b^i(j); r_a(i) := (j-1)/\omega;$
  end for
  $\chi := \chi - \iota$
end while
$U_a^p := \Sigma_{i=1}^m u_a^p(i); U_b^p := \Sigma_{i=1}^m u_b^p(i)$;

Theorem 3. *Pareto optimal results can be obtained by algorithm 1.*

Proof: Because algorithm 1 is based on the greedy algorithm, the utility of agent a is optimal without decreasing the utilities of agent b. According to the definition, algorithm 1 can realize Pareto optimal of agents. **End**.

Theorem 4. *The computational complexity of algorithm 1 is $O(\omega m \eta)$, $\eta = \chi/\iota$.*

Proof: Agent b will obtain all of resources in the worst case. Thus the inner nesting of 'While' cycle should be done ω times, 'For' cycle should be done m times, and the outer 'While' cycle should be done η times. Therefore, the computational complexity of algorithm 1 is $O(\omega m \eta)$. **End**.

Property 5. *The computational complexity of algorithm 1 is usually lower than that of Fatima et al.*

Proof: The computational complexity of the work of Fatima et al is $O(m\hat{\pi}l^3 t(h - t/2))$, where $t = min(2l-1, h)$, h is the maximum turn of negotiation. Thus it is $O(m\hat{\pi}l^3(2l-1)(h-l+0.5))$ i.e. $O(m\hat{\pi}l^4 h)$ when $h \geq (2l-1) \rightarrow t = 2l-1$. It is $O(m\hat{\pi}l^3h^2)$ when $h < (2l-1) \rightarrow t = h$. The computational complexity of algorithm 1 is $O(\omega m \eta)$. If we control the size of ω and η, the computational complexity of algorithm 1 is usually lower than that of Fatima et al. **End**.

3.2. Conditions of Acceptance for Offers

There are two conditions of acceptance for offers. One is that the utility of the current offer that the offering agent received is not smaller than that of the next his own counter-offer. Another is that the offering agent can not optimize the utility of own without decreasing the utility of the opponent, i.e. the current offer is a Pareto optimal offer.

Assumption 2. *Conditions of acceptance for offers*

Rule 6. *The utility of the current offer that the offering agent received is not smaller than that of the next his counter-offer.*

Rule 7. *The offering agent can not optimize the utility of own without decreasing the utility of the opponent.*

4. Negotiation with Complete Information

We assume that agent a and b know each other and the public information such as R, m, n, ρ is known by the two agents. We take the package deal procedure for this negotiation, then we can derive from the work of Fatima et al [5] that an agreement takes place at $t = 1$ for the package deal procedure, and the time taken to determine an equilibrium offer for $t = 1$ is $O(mn)$, where m is the number of resources and n is the deadline. We also know that the package deal procedure generates a Pareto optimal outcome.

We discuss the process of negotiation with complete information as follows. If negotiation reaches the deadline, then the agent whose turn it is takes everything and leaves nothing for its opponent. We then consider the preceding time periods ($t < n$), the offering agent will propose a counter-offer using greedy algorithm. Due to the complete information, each agent should consider the possibility of acceptance of the opponent. Furthermore, because of the discount of resources, each agent will not waste any time in order to avoid the loss of utility. Everyone knows that any more turn of negotiation means a loss of utility for both agents. So an agreement takes place at $t = 1$ for the package deal procedure.

So if agent b is the offering agent at $t = n$ turn, then agent b will take everything and leave nothing for agent a. It means that the utilities of agent b is $\rho^{n-1} \sum u_{b0}^i$, where $u_{b0}^i = k_b^i(1 - e^{-\lambda_b})$. At $t = n - 2$ turn, agent a uses greedy algorithm to generate optimal offer. This offer should maximize the utilities of agent a, as long as satisfying the requirement of agent b' utility, i.e. $\rho^{n-1} \sum u_{b0}^i$. So we can obtain the followings.

$\rho^{n-1} \sum u_{b0}^i = \rho^{n-2} \sum u_b^i$,
$\Rightarrow \rho \sum u_{b0}^i = \sum u_b^i$,
$\Rightarrow \rho \sum k_b^i(1 - e^{-\lambda_b}) = \sum k_b^i(1 - e^{-\lambda_b x_b^i})$,
$\Rightarrow \rho(1 - e^{-\lambda_b}) = \sum k_b^i(1 - e^{-\lambda_b x_b^i})$,

Now we can use a greedy algorithm to generate x_b^i, x_a^i. Because of complete information, agent b can not find a better outcome than this offer. So this offer is the offer at $t = 1$ turn for agent a.

If agent a is the offering agent at $t = n$ turn, then we can use a similar way to generate optimal offer at $t = 1$ turn.

Example 2. if $m = 2, n = 2, k_a^1 = 0.4, k_a^2 = 0.6, k_b^1 = 0.5, k_b^2 = 0.5, \lambda_a = 2, \lambda_b = 3, \rho_1 = \rho_2 = 0.5$, then

At $t = 2$ turn, agent b will take everything, then the utility of agent b is

$\rho(k_b^1(1-e^{-\lambda_b})+k_b^2(1-e^{-\lambda_b})) = 0.5*(0.5*(1-e^{-3})+0.5*(1-e^{-3})) = 0.5*(1-e^{-3}) = 0.475$.

Now we take greedy algorithm to compute x_b^i, where

$U_b^I = \rho(\sum k_b^i(1-e^{-\lambda_b})) = \rho(k_b^1(1-e^{-\lambda_b})+k_b^2(1-e^{-\lambda_b}))$.

We find that $\rho(1-e^{-\lambda_b}) = \sum k_b^i(1-e^{-\lambda_b x_b^i})$,

$\Rightarrow 0.5(1-e^{-3}) = 0.5(1-e^{-3x_b^1})+0.5(1-e^{-3x_b^2})$,

$\Rightarrow e^{-3x_b^1}+e^{-3x_b^2} = 1+e^{-3}$, where $\theta_1=\theta_2$.

So we can find that $x_b^1=0.258, x_b^2=0.177$ using the proposed greedy algorithm. Here $\theta_1=3.81=\theta_2, u_b^1=0.2694, u_b^2=0.206, u_a^1=0.31, u_a^2=0.48$. So $u_b^1+u_b^2=0.475=\rho(k_b^1(1-e^{-\lambda_b})+k_b^2(1-e^{-\lambda_b}))$.

Table 2. The offer of negotiation with complete information

Turn	x_a^1	x_a^2	x_b^1	x_b^2	u_a^1	u_a^2	u_b^1	u_b^2	U_a	U_b	θ_1	θ_2
2	0	0	1	1	0	0	0.475	0.475	0	0.475	0.093	0.062
1	0.742	0.823	0.258	0.177	0.309	0.484	0.269	0.206	0.79	0.475	3.81	3.81

5. Negotiation under Incomplete Information

We propose a negotiation model with two phases. There are two aims in the first phase. We use the strict alteration [2] where resources are taken as indivisible units and two agents alter to choose one resource to reach a preliminary agreement. At the same time, we can obtain some useful information about the opponent's preferences reasoned by two rules derived from the strict alteration. Sequentially, we take resources as divisible units and make tradeoff between two agents to get a Pareto optimal result in the second phase.

5.1. The First Phase

We cite the approach of strict alteration [2] in the first phase. Each agent will always choose the resource with the biggest preference in the remainder resources.

First phase:

Step 1: A random device chooses one of the two agents and marks this agent as s. Denote the set of resources yet to be negotiated by G. Initially, $G = R$.

Step 2: Now, s will choose one of the remaining resources $r \in G$, r is allocated to s.

Step 3: Mark the other agent as s and update G to $G-\{r\}$. If $|G| >= 1$ return to Step 2, otherwise stop.

If agent a is the first agent to choose resources and m is assumed to be even, then the following sequences are obtained. $x_a^1, x_a^2, \ldots, x_a^{m/2} \& x_b^1, x_b^2, \ldots, x_b^{m/2}$ (if m is odd, $x_a^{(m+1)/2}$ is added).

We can obtain a preliminary agreement in the first phase. Two agents allocate all resources where each resource is regarded as indivisible one. At the same time, this agreement is the base of the second phase where a greedy algorithm runs.

Through analyzing the above two sequences, agent a can obtain some useful information of agent b. Particularly, two rules of preferences of agent b can be obtained.

Agent a and b rank the resources with the preference. The resource with biggest preference of agent a is denoted by r_a^1, the resource with the ith big preference of agent a is denoted by r_a^i. The same is done for agent b. So agent a and b can obtain the following sequence.

$r_a^1, r_a^2, \ldots, r_a^m$ and $x_b^1, x_b^2, \ldots, x_b^m$

Example 3. For example, we assume that agent a and b alter to choose 6 resources $\{r_1, r_2, r_3, r_4, r_5, r_6\}$. We assume the following 4 cases illustrated in table 3. Furthermore, the results of choices of agent a and b are illustrated in table 4.

Table 3. The rank of the resources

Cases	r_a^1	r_a^2	r_a^3	r_a^4	r_a^5	r_a^6	r_b^1	r_b^2	r_b^3	r_b^4	r_b^5	r_b^6
Case 1	r_1	r_2	r_3	r_4	r_5	r_6	r_6	r_5	r_4	r_3	r_2	r_1
Case 2	r_1	r_2	r_3	r_4	r_5	r_6	r_1	r_2	r_3	r_4	r_5	r_6
Case 3	r_1	r_2	r_3	r_4	r_5	r_6	r_2	r_1	r_3	r_4	r_5	r_6
Case 4	r_1	r_2	r_3	r_6	r_5	r_4	r_1	r_3	r_5	r_4	r_2	r_6

Table 4. The results of choices of agent a and b

Cases	x_a^1	x_a^2	x_a^3	x_b^1	x_b^2	x_b^3
Case 1	r_b^6	r_b^5	r_b^4	r_b^1	r_b^2	r_b^3
Case 2	r_b^1	r_b^3	r_b^5	r_b^2	r_b^4	r_b^6
Case 3	r_b^2	r_b^3	r_b^5	r_b^1	r_b^4	r_b^6
Case 4	r_b^1	r_b^5	r_b^6	r_b^2	r_b^3	r_b^4

Rule 8. *x_b^i may be r_b^i to r_b^{2i}.*

Rule 9. *x_a^i may be r_b^i to x_b^m.*

Agent a should always choose the resource remained with the biggest r_a^i, no matter whether it is that of agent b. whether agent a has chosen agent b's expected resource is the key problem because agent a always be ahead of agent b and can affect agent b's selection. Three cases exist according to the relationship between agent a and agent b.

If two agents do not conflict with each other anytime, i.e. agent a and b can always choose the expected resources. So x_b^i should always be r_b^i, and x_a^i may be $r_b^{m/2}$ to x_b^m. Let's see case 1, x_b^1 is r_b^1, x_b^2 is r_b^2, and x_b^3 is r_b^3. While x_a^1 is r_b^6, x_a^2 is r_b^5, x_a^3 is r_b^4.

If two agents conflict with each other all along, i.e. agent b can not choose the expected resources all the while. What agent a chooses is the expected resource of agent b. It means that x_a^i should be r_b^i to r_b^{2i-1}. What agent b can choose is chosen by agent a. So agent b can only choose r_b^{2i}. So x_b^i should always be r_b^{2i}. Let's see case 2, x_a^1 is r_b^1, x_a^2 is r_b^3, x_a^3 is r_b^5. So agent b can only choose the next resource, then x_b^1 is r_b^2, x_b^2 is r_b^4, and x_b^3 is r_b^6.

If two agents partly conflict with each other, then x_b^i may be in $\{r_b^i \dots r_b^{2i}\}$, and x_a^i may be in $\{r_b^i \dots x_b^m\}$. Let's see case 4, x_a^1 is r_b^1, x_a^2 is r_b^5, x_a^3 is r_b^6. So x_b^1 is r_b^2, x_b^2 is r_b^3, and x_b^3 is r_b^4.

Example 4. Let's consider such a setting, agent a and b negotiate on four divisible resources $R = \{r_1, r_2, r_3, r_4\}$. Agent b is well known to be one of three possible classes of the opponents whose preferences and λ^b are shown in the followings.

Table 5. Information state of agents($P^b =< 0.5, 0.3, 0.2 >$)

Items	r_1	r_2	r_3	r_4	λ	Rank of resources
k_a	0.4	0.25	0.15	0.2	2	$r_1 \succeq r_2 \succeq r_4 \succeq r_3$
$k_b^{C_1}$	0.2	0.3	0.15	0.35	2.5	$r_4 \succeq r_2 \succeq r_1 \succeq r_3$
$k_b^{C_2}$	0.4	0.25	0.15	0.2	2	$r_1 \succeq r_2 \succeq r_4 \succeq r_3$
$k_b^{C_3}$	0.1	0.4	0.3	0.2	3	$r_2 \succeq r_3 \succeq r_4 \succeq r_1$

$k_a =< 0.4, 0.25, 0.15, 0.2 >, \lambda_a = 2$; $k_b^{C_1} =< 0.2, 0.3, 0.15, 0.35 >, \lambda_b^{C_1} = 2.5$; $k_b^{C_2} =< 0.4, 0.25, 0.15, 0.2 >, \lambda_b^{C_2} = 2$; $k_b^{C_3} =< 0.1, 0.4, 0.3, 0.2 >, \lambda_b^{C_3} = 3$; $P^b =< 0.5, 0.3, 0.2 >$.

Where, $k_b^{C_i}$ denotes the preference vector when agent b is C_i class, $\lambda_b^{C_i}$ denotes the firing rate when agent b is C_i class.

If $< x_a^1 = r_1, x_b^1 = r_4, x_a^2 = r_2, x_b^2 = r_3 >$ is obtained through the approach of strict alternation in the first phase. We can reason that only $k_b^{C_1}$ satisfies the fact that r_4 is ranked as the first or second order of agent b. Therefore agent b must be C_1 class. The approach is generally feasible when there are a few classes of agents.

Therefore, agent b obtains all of $r_3 \& r_4$ in the first phase through the approach of strict alternation. The initial utilities of agent b are 0.45896 while those of agent a are 0.56203. **End**.

Alternatively, if agent b firstly chooses resources and $< x_b^1 = r_4, x_a^1 = r_1, x_b^2 = r_2, x_a^2 = r_3 >$ is obtained. We also obtain that only $k_b^{C_1}$ satisfies the fact r_4 is ranked as the first order. Therefore agent b must be C_1 class. Agent b obtains all of $r_4 \& r_2$ in the first phase of negotiation trough the approach of strict alternation.

Based on the above rules, we can obtain some useful information which will help us in the next phase. If there is only one possible class, then agent b must

Table 6. Strict alteration of the first phase

The first agent to offer	x_b^1	x_a^1	x_b^2	x_a^2	$k_b^{C_i}$	r_a	r_b	U_a^I	U_b^I
a	r_1	r_4	r_2	r_3	$k_b^{C_1}$	$r_1\&r_2$	$r_3\&r_4$	0.56203	0.45896
b	r_4	r_1	r_2	r_3	$k_b^{C_1}$	$r_1\&r_3$	$r_2\&r_4$	0.47557	0.59664

be it; Otherwise, we can take advantage of probability of opponents to select the type with the maximal probability. So not only preliminary agreement but also preferences and firing rate of opponents can be obtained in the first phase.

5.2. The Second Phase

After phase 1, we can obtain a preliminary agreement and some useful information. Sequentially, we enter the second phase to obtain a Pareto optimal result.

Based on the agreement of phase 1, if agent b's utility requirement is satisfied, then agent b should give this offer to agent a. Otherwise agent b should firstly satisfy its own requirement and give its new offer to agent a.

When agent a receives an offer from agent b, it should compute its own utility using algorithm 1. If agent a's utility requirement can be satisfied by this offer, then agent a should send agent b the new offer. Because the new offer would satisfy agent b's requirement, agent b should accept this new offer and negotiation succeeds. If agent a's utility requirement can not be satisfied by this offer, then agent a would not consider the requirement of agent b and only satisfies its own utility requirement, then it provides agent b the new offer based on its own preferences and utility function.

When agent b receives an offer from agent a, it should compute its own utility. If agent b's utility requirement can be satisfied by the counter-offer, agent b will accept the counter-offer. Negotiation succeeds. If it is not, then return to the beginning step of ”Otherwise”.

We give the following flow to understand the above operations.

Second phase:

Step 1: Set the result of the first phase as the initial allocation of resources. Choose one of the two agents and mark it as a, the other agent is marked as b.

Step 2: If agent b's utility requirement can be satisfied in the first phase, then agent b gives the allocation of the first phase to agent a.

Step 3: Else agent b provides the offer based on its own preferences and utility function, where its utility requirement can be satisfied. end if

Step 4: agent a receives the offer from agent b, then to compute its utility using algorithm 1.

Step 5: If agent a's utility requirement can be satisfied by this offer of algorithm 1, then agent a gives the optimal counter-offer to agent b. Agent b should accept the counter-offer because its own utility requirement can be satisfied. Negotiation succeeds.

Step 6: Else agent a provides the offer based on its own preferences and utility function, where its utility requirement can be satisfied.

Step 7: If agent b's utility requirement can be satisfied by the counter-offer, agent b will accept the counter-offer. Negotiation succeeds.

Step 8: Else go to step 3.

6. Experimental Evaluation

6.1. Optimal Performance of Algorithm 1

Recall example 4, agent a and b negotiate on 4 resources using this model. After the first phase, agent a obtained the following optimal results by algorithm 1.

Table 7. The results of experiments where agent a firstly chooses resources

Utility	agent a	agent b	x
r_1	0.33647	0.036254	0.92
r_2	0.19077	0.15102	0.72
r_3	0.11784	0.065594	0.77
r_4	0.14439	0.2077	0.64
Initial sum	0.56203	0.45896	1,1,0,0
Optimal sum	0.78948	0.46057	0.92,0.72,0.77,0.64

After optimizing in the second phase of negotiation, agent b obtains 0.92 proportion of r_1, 0.72 proportion of r_2, 0.77 proportion of r_3, 0.64 proportion of r_4. Sequentially, the optimal utility of agent b is 0.46057 and that of agent a is 0.78948 through the second phase. Experimental results show that the utilities of agent a significantly excel the preliminary agreement without decreasing the utilities of agent b. We practically improve the utilities of agent a by 40.5% (0.78948:0.56203)and that of agent b by 0.35% (0.46057:0.45896). The results of experiments are shown in Table 7.

Alternatively, if agent b firstly chooses resources. We improve the utilities of agent a by 53.8% (0.73155:0.47557) and that of agent b by 0.9% (0.60193:0.59664). Experimental results are shown in Table 8.

6.2. Influence of λ_a and λ_b

If we change the value of λ_a and λ_b from 0.2 to 20, we get the results illustrated in Fig. 5, where U_a^o denotes the optimized utilities of agent a. U_b^o denotes the

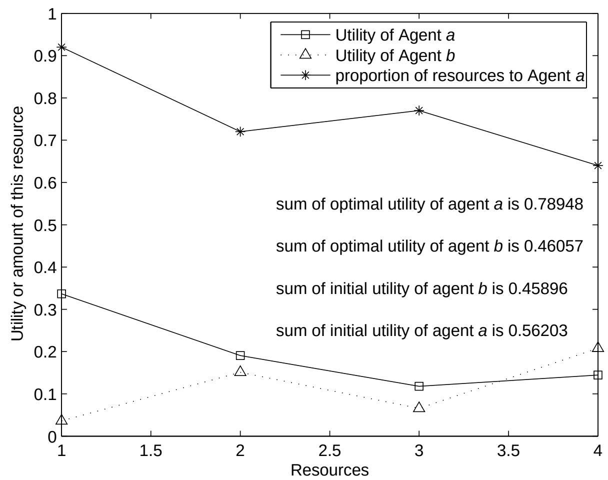

Fig. 4. Optimized results of algorithm 1

Table 8. The results of experiments where agent b firstly chooses resources

Utility	agent a	agent b	x
r_1	0.31924	0.078694	0.8
r_2	0.1747	0.18964	0.6
r_3	0.10829	0.089015	0.64
r_4	0.12931	0.24458	0.52
Initial sum	0.47557	0.59664	1,0,1,0
Optimal sum	0.73155	0.60193	0.8,0.6,0.64,0.52

optimized utilities of agent b. $U^o_{a\&b}$ denotes the optimized utilities of agent a and agent b.

We can find that when the value of $\lambda_a = \lambda_b$ is small, the effect of optimization is mainly on agent a. If $\lambda_a = \lambda_b < 4$, few effect is added on agent b. If $\lambda_a = \lambda_b > 5$, the utilities of agent a are nearly saturated. At the same time, the utility of agent b is optimized greatly. So the effect of optimization is perfect. However, if $\lambda_a = \lambda_b > 10$, the utilities of agent a and b are both nearly saturated. So the effect of optimization is not well.

6.3. Influence of Different δ_1 and δ_2

If we change δ_1 from 0.0001 to 0.05, the results of experiments are shown in Fig. 6. In which, a_i denotes the initial utilities (not optimized) of agent a, a_o denotes the optimized utilities of agent a, b_i denotes the initial utilities of agent a, b_o denotes the optimized utilities of agent b.

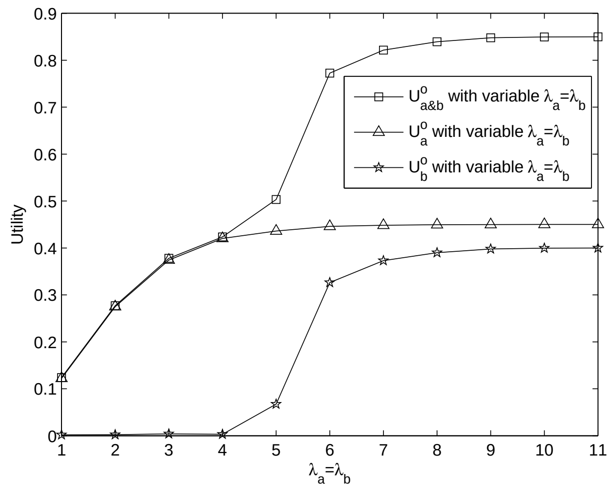

Fig. 5. Influence of different $\lambda_a = \lambda_b$ to the utilities of agent a and b

We can find that the optimal utilities of agent b becomes bigger, that of agent a becomes smaller, and the social welfare of optimal utility of agent a and b becomes also bigger.

6.4. Influence of Various Kinds of Agents

If various kinds of agents participate the negotiation, we can get the average optimized efficiency in a small range. We take 18 various kinds of agents and these results are illustrated in Fig. 7, where U_a^I denotes the initial utilities of agent a, U_a^o denotes the optimized utilities of agent a, U_b^I denotes the initial utilities of agent b, U_b^o denotes the optimized utilities of agent b.

We can find that the average utilities of agent a is improved by 59.56%. The average utilities of agent b is improved by 0.85%. The average utilities of agent a and b is improved by 28.95%.

The utilities of agent a and b is usually bigger than 1 because the parameter λ is enough big and the firing rate is enough high and it realizes the result of win-win.

6.5. Comparison With Linear Utility Functions

If we compare the optimized results of nonlinear utility functions with those of linear utility functions, we can find that the former is higher than the latter. The reason may be that the former considers not only the preferences of agents, but also the quantity of resources. Thus the former is more possible to get

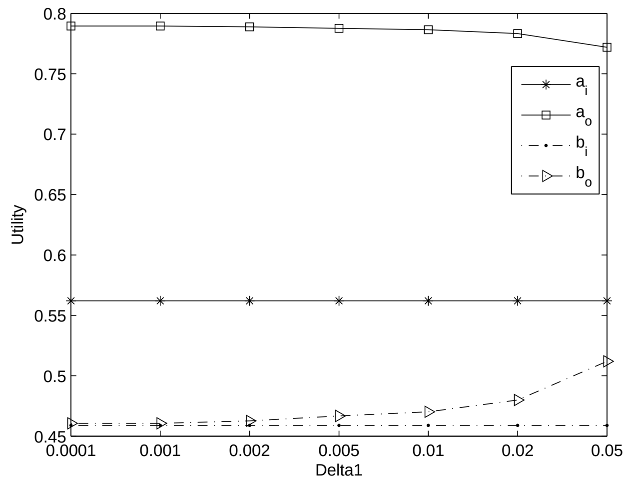

Fig. 6. Influence of different δ_1 to the utilities of agent a and b

more utilities than the latter. Table 8 and 9 show the comparison of optimized efficiency among the work of Fatima, Faratin and ours. We can discover the above conclusion from these tables. More results of comparison are illustrated in section 6.6.

Table 9. Comparison of the results of optimization between linear utility function and nonlinear utility function where agent a firstly chooses resources

Utility	agent a	agent b	Sum of agent $a\&b$	Percent of optimization
Initial	0.56203	0.45896	1.02099	-
Optimization of Fatima	0.7092	0.45896	1.16816	14.41444
Optimization of Faratin	0.56203	0.635564	1.197594	17.29733
Optimization of ours	0.78948	0.46057	1.25005	22.43509

6.6. Comparison With the Work of Fatima

We compare experimental results of algorithm of Fatima et al and that of algorithm 1. The result is illustrated in Fig. 8, where U_a^I denotes the initial utilities of agent a, U_a^o denotes the optimized utilities of agent a using the algorithm of this paper, U_{aF}^I denotes the initial utilities of agent b using the algorithm of Fatima et al, U_{aF}^o denotes the optimized utilities of agent a using the algorithm of Fatima et al.

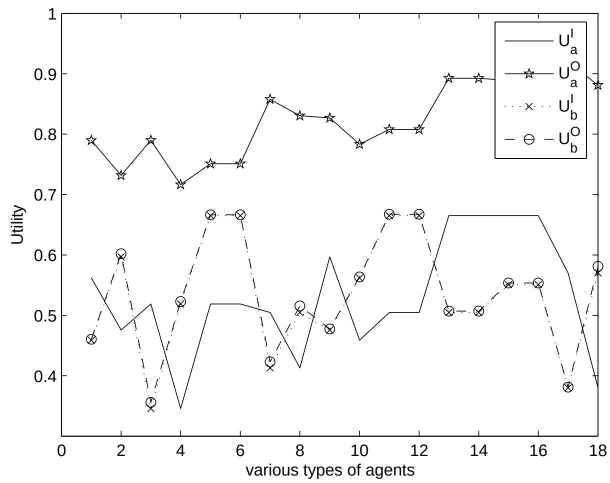

Fig. 7. Influence of various kinds of agents to the utilities of agent a and b

Table 10. Comparison of the result of optimization between linear utility function and nonlinear utility function where agent b firstly chooses resources

Utility	agent a	agent b	Sum of agent $a\&b$	Percent of optimization
Initial	0.47557	0.59664	1.07221	-
Optimization of Fatima	0.594467	0.59664	1.191107	11.08893
Optimization of Faratin	0.47557	0.72443	1.2	11.91837
Optimization of ours	0.73155	0.60193	1.33348	24.36743

There are 18 cases if agent a and b are different kinds and alternatively firstly choose resources in the first phase. The average initial utilities of agent a in all 18 cases using algorithm 1 is 0.529661 and that of optimized utilities is 0.822555, therefore algorithm 1 improves utilities of agent a by 59.56%. While using methods of Fatima et al, these data are 0.580556, 0.583611, 0.55% respectively. Experimental results show algorithm 1 takes an advantage over the work of Fatima et al.

7. Background and Related Work

In recent years, researches on agent multi-issue negotiation with incomplete information attract more and more attention. Generally speaking, there are simultaneous procedure, sequential procedure, and package deal procedure approaches. However, only package deal procedure can ensure Pareto optimal solution indicated by Fatima et al [5]. Therefore, we only introduce the approach of package deal procedure to ensure Pareto optimal results in this paper.

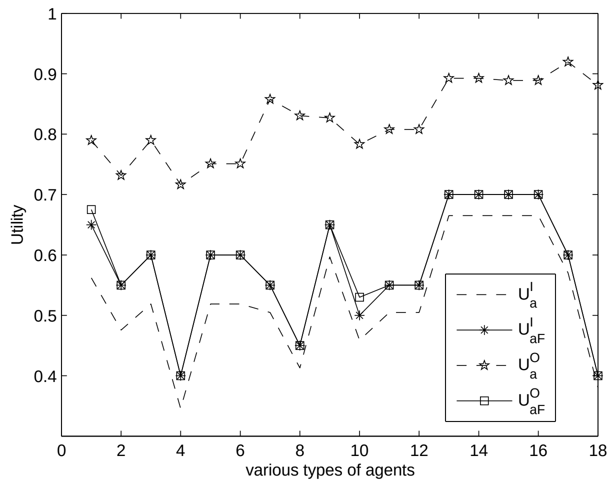

Fig. 8. Comparison with the work of Fatima in the utilities of agent a

Negotiation on multiple resources is always a challenging task. Dunne et al [15] study automatic contract negotiation in e-commerce and e-trading environments, they consider the computational complexity of a number of natural decision problems of one-resource-at-a-time to construct a mutually beneficial optimal reallocation by trading resources. An et al [13] present a proportional resource allocation mechanism and give a game theoretical analysis of the optimal strategies with incomplete information. Saha et al [2] present a protocol for negotiation on multiple indivisible resources which can be used by rational agents to reach efficient outcomes through searching negotiation tree. Chevaleyre et al [3] try to balance efficiency and fairness requirements to set up a distributed negotiation framework which will allow a group of agents to reach an allocation of goods that is both efficient and envy-free. Lin et al [18] conduct experiments on the allocation of circumstance of work and the World Health Organization Framework Convention on Tobacco Control. However, all of these existed researches have not given clear utility function, instead, utility value is given in advance or resources are directly taken as utilities. Unfortunately, there are many cases where the utility function is nonlinear. As well as Wooldridge [9] points out utility is not money but it is a useful analogy and he gives a typical relationship between utility and money like saturated nonlinear function.

A number of researches are dedicated to coping with incomplete information by learning preferences of opponents. Luo et al [1, 19] learns preferences of opponents based on knowledge models and through ”default then adjust” approach. Lin et al [18] determine kinds of agents through offers in the process of negotiation based on the approach of naive Bayes class. The principle of our work is similar with the approach of Lin et al that we determine types of opponents by reasoning from the information obtained in the first phase.

A Pareto optimal result is another important aim in multi-issue negotiation and some effective researches are done recently. Faratin et al [16] propose to maximize the utilities of opponents based on the same utilities of oneself to improve acceptive possibility of opponents. While Fatima et al [5] propose a similar approach to maximize the utilities of oneself based on the same utilities of opponent. The approach of our work is similar with that of Fatima et al but different from the definition and the treatment of utility functions.

Nonlinear utility functions were focused in the past years. Fatima et al [22] analyze bilateral multi-issue negotiation involving nonlinear utility functions.They show that it is possible to reach Pareto-optimal agreements by negotiating all the issues together, and that finding an equilibrium is not computationally easy if the agents' utility functions are nonlinear. They investigate two solutions: approximating nonlinear utility spaces with linear functions, and using a simultaneous procedure where the issues are discussed in parallel but independently of each other. They show that the equilibrium solution can be computed in polynomial time. However, their work is focused on symmetric negotiations where the agent's preferences are identically distributed, and the utility functions are separable in nonlinear polynomials of a single variable.

Vazirani et al [11] investigated some issues of computational complexity on market equilibrium under separable, piecewise-linear, concave utilities. They consider Fisher and Arrow-Debreu markets under additively separable, piecewise-linear, concave utility functions and obtain the following results. For both market models, if an equilibrium exists, there is one that is rational and can be written using polynomially many bits. There is no simple necessary and sufficient condition for the existence of an equilibrium. Under standard (mild) sufficient conditions, the problem of finding an exact equilibrium is in PPAD for both market models. Finally, they prove that under these sufficient conditions, finding an equilibrium for Fisher markets is PPAD-hard.

Complex nonmonotonic preference spaces were investigated by various researchers such as Ito et al. [21], Marsa-Maestre et al. [23, 24], Lopez-Carmona et al. [12], where Lopez-Carmona et al. proposed a region-based automated multi-issue negotiation protocol (RBNP), which is built upon a recursive non-mediated bargaining mechanism. The non-monotonic negotiation scenarios are created using an aggregation of Bell functions. In contrast to prior researches, which usually assume that agents have relatively simple preferences on the issues (e.g., can be characterized by strictly concave utility functions), they make a more general assumption that the preference of each agent can be non-monotonic and non-differentiable.

8. Conclusions

This paper proposed an improved nonlinear utility function motivated by firing function of artificial neuron of artificial neural network. The proposed utility function follows the law of diminishing marginal utility. The negotiation model presented in this paper can deal with the instances of incomplete information, and Pareto optimal solutions will be obtained as a result. Experimental results show that the efficiency of optimal approach has an advantage over prior work.

The future work may improve the first phase of negotiation to reach a more rational preliminary agreement. A new optimal approach is also required to maximize the utilities of two agents, as well as maximizing the utilities of agent a.

Acknowledgments. The authors would like to thank School of Computer Science of Yantai University in China. In the laboratory of Intelligence Information Processing, the authors have received lots of enthusiasm and help without any personal motivations.

This work is partly supported by the Major State Basic Research Development Program of China under Grant (2007CB307100), the National Science Foundation of China (61170224, 60973075), the Shandong Natural Science Foundation (ZR2011FL018, 2012GGB01017, ZR2012FL07), and the Technology Research Program of Education Commission of Shandong Province (J11LG35, J10LG27).

References

1. X. Luo, N. R. Jennings, N. Shadbolt, *Acquiring user tradeoffs and preferences for negotiating agents: a default then adjust method*. International Journal of Human Computer Studies, (2006).
2. Sabyasachi Saha, Sandip Sen, *An Efficient Protocol for Negotiation over Multiple Indivisible Resources*. IJCAI'07, India, 1494-1499, (2007).
3. Yann Chevaleyre, Ulle Endriss, Sylvia Estivie, NicolasMaudet, *Reaching Envy-free States in Distributed Negotiation Settings*. IJCAI'07, India, 1239-1244, (2007).
4. S. S. Fatima, M. Wooldridge, N. R. Jennings, *An agenda-based framework for multi-issue negotiation*. Artificial Intelligence, 152(1), pp. 1-45, (2004).
5. S. S. Fatima, M. Wooldridge, N. R. Jennings, *Multi-Issue Negotiation with Deadlines*. Journal of Artificial Intelligence Research (JAIR), 27, pp. 381-417, (2006).
6. Tong Xiangrong, Zhang wei. *Agent multi-issue negotiation with cases*. The 5th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'08), Jinan, 96-100, (2008).
7. Tong Xiangrong, Houkuan Huang, Zhang wei. *A Case based Agent Multi-issue Negotiation Model*. Jounal of Researches and Development of Computer, 46(9): 1508-1514(in Chinese), (2009).
8. Tong Xiangrong, Guo Liang. *Learning opponent's indifference curve in multi-issue negotiation combining rule based reasoning with bayesian classifier*. The proceedings of 2010 sixth international conference on natural computation(ICNC'10), Yantai, 2916-2920, (2010).
9. Michael Wooldridge, *An introduction to multiAgent systems*. England: John Wiley Sons, Inc, (2002).
10. J.W. Pratt, *Risk Aversion in the Small and in the Large*, Econometrica 32:122-136, (1964).
11. Vijay V.Vazirani, Mihalis Yannakakis. *Market Equilibrium under Separable, Piecewise-liner, Concave Utilities*. Journal of ACM, 58(3):10, (2011).

12. Miguel A. Lopez-Carmona, Ivan Marsa-Maestre, Enrique De La Hoz, and Juan R. Velasco. *A Region-based Multi-issue Negotiation Protocol for Nonmonotonic Utility Spaces*. Computational Intelligence, 27(2): 166-217, (2011).
13. Bo An, Chunyan Miao, Zhiqi Shen, *Market Based Resource Allocation with Incomplete Information*. IJCAI'07, India, 1193-1198, (2007).
14. R. M. Coehoorn, N. R. Jennings, *Learning an opponent's preferences to make effective multi-issue negotiation tradeoffs*. Proc. 6th Int Conf. on E-Commerce, Delft, The Netherlands, 59-68, (2004).
15. Paul E. Dunne, Michael Wooldridge, Michael Laurence, *The complexity of contract negotiation*. Artificial Intelligence 164, pp. 23-46, (2005).
16. P. Faratin, C. Sierra, N. R. Jennings, *Using similarity criteria to make trade-offs in automated negotiations*. Artificial Intelligence 142 (2), pp. 205-237, (2002).
17. Sarit Kraus, Penina Hoz-Weiss, Jonathan Wilkenfeld, David R. Andersen Amy Pate, *Resolving crises through automated bilateral negotiations*. Artificial Intelligence, 172(1): 1-18, (2008).
18. R. Lin, S. Kraus, J. Wilkenfeld, J. Barry, *Negotiating with Bounded Rational Agents in Environments with Incomplete Information Using an Automated Agent*. Artificial Intelligence 172(6-7), pp. 823-851, (2008).
19. X. Luo, N. R. Jennings, N. Shadbolt, H. Leung, J. H. Lee, *A fuzzy constraint based model for bilateral multi-issue negotiations in semi-competitive environments*. Artificial Intelligence 148 (1-2), pp. 53-102, (2003).
20. Sarvapali D. Ramchurn, Carles Sierra, Llułs Godo, Nicholas R. Jennings, *Negotiating using rewards*. Artificial Intelligence 171, 805-837, (2007).
21. Takayuki Ito, Hiromitsu Hattori and Mark Klein, *Multi-issue Negotiation Protocol for Agents: Exploring Nonlinear Utility Spaces*. IJCAI'07, pp. 1347-1352, (2007).
22. Fatima, S.,M.Wooldridge, and N. Jennings. *An analysis of feasible solutions for multi-issue negotiation involving nonlinear utility functions*. In AAMAS'09: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1041-1048, (2009).
23. Marsa-Maestre, I., M.A. Lopez-Carmona, J.R. Velasco, and E. De La Hoz. *Effective bidding and deal identification for negotiations in highly nonlinear scenarios*. In Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS'09), Budapest, Hungary, 1057-1064, (2009).
24. Marsa-Maestre, I., M.A. Lopez-Carmona, J.R. Velasco, T. Ito, M. Klein, and K. Fujita. *Balancing utility and deal probability for auction-based negotiations in highly nonlinear utility spaces*. In 21st International Joint Conference on Artificial Intelligence (IJCAI'09), Pasadena, California, 214-219, (2009).

Xiangrong Tong, Ph. D. Associate Professor, Member of China Computer Federation, School of Computer Science, Yantai University, Yantai, China. His interests of researches include distribute artificial intelligence, multi-agent systems and data mining.

Wei Zhang, Professor, Senior Member of China Computer Federation, School of Computer Science, Yantai University, Yantai, China. His interests of researches include distribute artificial intelligence and multi-agent systems.

Houkuan Huang, Professor and Ph. D. supervisor, Senior member of China Computer Federation, Beijing Jiaotong University, Beijing, China. His interests of researches include artificial intelligence, data mining, and machine learning.

Received: March 6, 2012; Accepted: December 3, 2012.