http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 10 (11): 2014-2023, 2011
ISSN 1812-5638 / DOI: 10.3923/itj.2011.2014.2023

Study of Automated Trust Negotiation Mechanism Based on Cache Sequence Game in P2P Environment

[1]Chunzhi Wang, [2]Ruoxi Wang, [1]Hongwei Chen and [1]Ya Huang
[1]School of Computer Science, Hubei University of Technology, China
[2]Department of Computer Science, Huazhong Normal University, China

Abstract: The process of trust negotiation is the one of the resource requester and resource provider exposing the certificate set and strategy set with each other. The network authentication model based on trust negotiation fulfills the P2P network security, privacy, efficiency and success. The negotiation model will be constructed by the trust game tree. When the negotiation succeeds, cache the certificate sequence and directly show the sequence to complete the negotiation in the next negotiation. But the complete caching sequence will lead a heavy burden and cause safety problems to the system, splitting negotiation process into multi-step and caching part of sequence to improve overall performance. This study builds the caching sequence model based on mixed static game theory and simulates the situation of the caching sequence on partial by introducing the ratio factor. Finally, verify the assumption via the simulation tools named Gambit.

Key words: P2P, trust negotiation, mixed game, trust game tree, caching sequence, ratio factor

INTRODUCTION

The strategies (Traum *et al*., 2008; Skogsrud *et al*., 2009) of trust negotiation (Li *et al*., 2009) are two kinds: Positive strategy and cautious strategy. The positive strategy discloses all the certificates in one time in order to attain trust. Because it has counteracted the income the communication overhead brings about that the privacy loss that disclosing the certificates indiscriminatively brings about, most of existing negotiation framework uses the cautious strategy (Zhang and Winslett, 2008), such as Trust-X, TrustBuilder, PolicyMaker and so on. Every structure all provided detailed certification, rules for policy making and flow chart for negotiation. In addition, it also advances mechanism of caching sequence to save time cost of negotiation. Complete caching sequence would store all the producing certification sequence in the process of negotiation, at the moment of gaining time yield, consumption of saving resource would trigger a series of safety problems.

The two nodes exchange the set of access control strategy (Lu and Liu, 2009; Ardagna *et al*., 2008) and disclose the certificates gradually to unlock the resources. This kind of method might cause too communication overhead to reduce the efficiency of negotiation because of negotiating repeatedly, thus to import Trust Ticket and Cache Sequence to optimize. Traditional negotiation strategy constructs the negotiation tree according to resources or certificates as nodes and shows the relationship of disjunction and conjunction. The values of edges connecting nodes indicate the comprehensive income of the certificates of the nodes and the sub-nodes.

TRUST NEGOTIATION MODEL BASED ON THE EXTENSIVE GAME

Trust negotiation sequence: Figure 1 shows a typical process of cautious negotiation. Figure 2 describes a constructed negotiation tree for resource R and the tree is constructed in the process of the participators negotiating gradually.

In Fig. 2a, there are all the routes of the unlock-resource R, in which the square represents disjunction and the pentagon represents conjunction. The sides whose weight is 8 represent that they cannot obtain the certificates of the next node, so the pay expense of disclosing reaches of infinite. In Fig. 2a, there are some selectable negotiation routes (Chou *et al*., 2008) to be able to construct trust finally and the different routes have the different certificate sets of disclosing, on the basis of the value of utility of certificates to be able to obtain the different income. In Fig. 2b, it is the one, in order to get the resources, the requester needs to disclose

Corresponding Author: Hongwei Chen, School of Computer Science, Hubei University of Technology, China

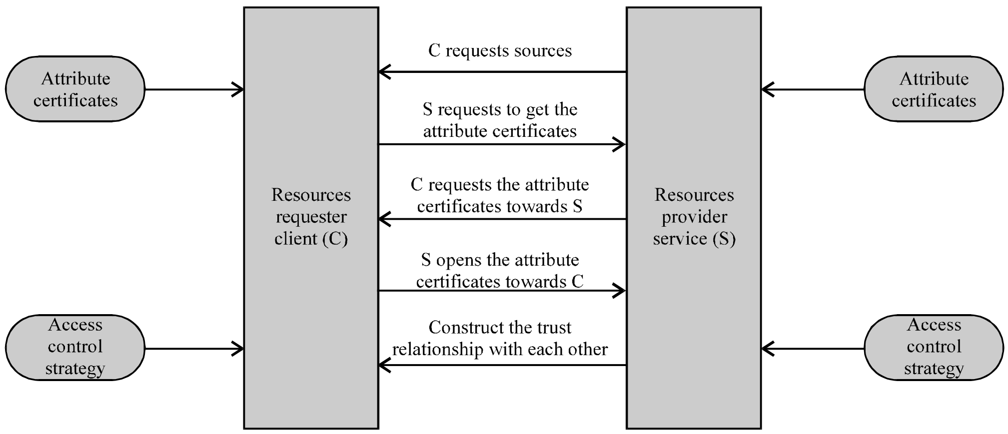

Fig. 1: The process of trust negotiation

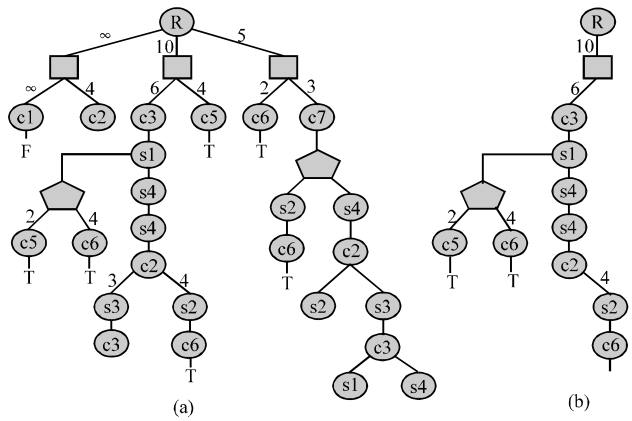

Fig. 2 (a-b): The process constructed by negotiation tree; (a) 2-1 and (b) 2-2

the following certificate set {C_2, C_3, C_5, C_6}. In order to conduct the requester's behavior, the provider should also disclose the following certificate set {S_1, S_2, S_4} which can just construct trust. This method, in the situation of a mass of certificates and the complication of access control strategy, may cause the space burden and accessing burden (Zhang *et al*., 2008) because of the excessiveness of nodes and may aggravate the corresponding pay expense because of the frequent negotiation process.

Model of game negotiation: The existing negotiation strategy is the process to construct the negotiation tree, by the certificates as nodes and by the process of unlock-certificates as sides. Here, another kind of process to construct the negotiation tree is put forward by the resource requester and provider as nodes and by the certificate set of disclosing as sides. This process is extremely similar as the extensive game with perfect information. Easily finding out, both who participate in the game can be abstracted by decision node and the certificate set of disclosing can be abstracted by branch. The process of negotiation is the one of constructing the expended game tree; via income function to calculate the participator's final effectiveness and to solve the sub game perfect Nash Equilibrium. The actual negotiation process is very complex and it can simplify in these condition of convention:

- The resource requestor C and the resource provider S are both completely rational
- The resource requestor C and the resource provider S will pursue the maximum income for themselves in the premise of insuring to construct trust
- The resource requestor C and the resource provider S can obtain counterpart's access control strategy via negotiation

To solve the sub game perfect Nash Equilibrium is to use the backward induction only if which insure that the participators are completely rational, the relatively optimal solution would be obtained and otherwise the final income is unsatisfactory. According to the agreement, it can be described as a tetrad as Γ = (P, T, Π, U) in the open network environment that to construct extended game.

In the tetrad, P indicates the set of game participators and in the game P = {Client, Service}; T indicates the action sequence of participators when the game terminates, Δ is the set of all sequences and the participators' action set defines as A(t) = {P (t, C_x)} after a moment t and Π indicates the function to determine the participator's action which needs that the other shows the certificate set:

$$\Pi\ (t)\begin{cases} C, if\{t=\varnothing\} \\ C, if\{t=(t',S'x),(t'\in\Delta,S'x\in Sx)\} \\ S, if\{t=(t',C'x),(t'\in\Delta,C'x\in Cx)\} \end{cases} \quad (1)$$

and U indicates the final state income function of game and the income to the extent influence the choice of the participators.

In order to save the communication burden of negotiation, at the beginning with negotiation, both the nodes is to change all the access control strategy of certificates and then both of the negotiation create the negotiation tree and calculate the income in local. Because both have the same knowledge and information in the game, the minimum certificate set of disclosing can be uniquely confirmed. Although, in this case the privacy of access control strategy would be lost, the times and the communication pay expense can be decreased. In addition, compare with the private burden caused by disclosing the certificates, the cost to disclose the access control strategy is insignificant.

According to the calculating formula mentioned above, can obtain the model of trust negotiation based on the extensive game shown in the Fig. 3 as follows:

In Fig. 3, the process of trust game is as follows:

Step 1: The resource requestor C requests the resource R from the resource provider S

Step 2: The resource requestor C requests the access control strategy of the resource and the certificate from the resource provider S

Step 3: The resource provider S requests the access control strategy of the certificate from the resource requestor C

Step 4: The resource requestor C and the resource provider S exchange the access control strategy from each other

Step 5: The resource requestor C and the resource provider S construct the extensive game tree according to step 1-4 and solve the optimal action sequence to find the certificate set using the algorithm of the extensive sub game perfect Nash Equilibrium

Step 6: The resource requestor C and the resource provider S search the certificate set which is hold in local according to the step 5

Step 7: The resource provider S sends the selected attribute certificate set of the step 6 to the resource requestor C

Step 8: The resource requestor C sends the selected attribute certificate set of the step 6 to the resource provider

The income calculation of negotiation strategy: Calculating the income of negotiation strategy is to need to analyze the selected certificate set of the action of

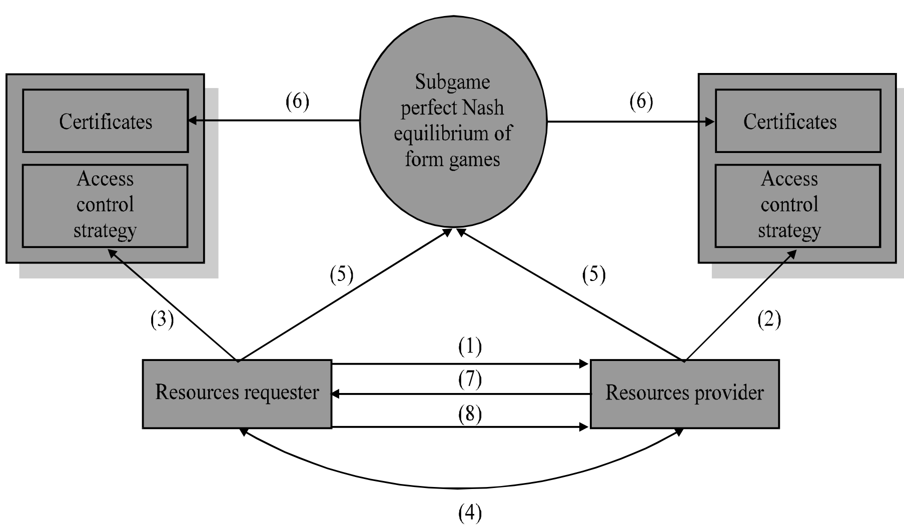

Fig. 3: The flow chart of trust game

participators and firstly is to need to obtain the value of the effectiveness of certificates. The methods of calculating the value of the effectiveness of certificates are many, such as the static one and the dynamical one. The static method can be determined at the initialization of the system according to the uniform regulation. This kind of method is simple but inaccurate, because the effectiveness of the same certificate might change as time goes by. This section put forward to a kind of method to calculate the effectiveness of certificates dynamically. If the resource provider has the certificate S_x, the more certificates the resource requestor need to supply in order to unlock S_x, the higher the value of S_x is and the more the loss to disclose the certificate for its own is. If the resource requestor has the certificate C_x, the more the times to need in the access control strategy of the resource provider, the more to earn for its own is. The value of the certificate of the resource provider for its own defines as U_s (C_x) and the value for the resource provider defines as U_s (C_x). In addition, the basic value of every certificate is **δ**, the basic of public certificate is 2 and the basic of non-public certificate is 3. The calculating formula of effectiveness of certificates is $U_s (C_x)-U_s (S_x) + \delta$ or $U_C (C_x)-U_c (S_x)+S$.

Client	Service
$P_{C1} = C_2 \leftarrow S_2 \vee S_3$	$P_{S1} = R \leftarrow (C_1 \wedge C_2)\ (C_3 \wedge C_5)\ (C_6 \wedge C_7)$
$P_{C2} = C_3 \leftarrow S_1 \vee S_4$	$P_{S2} = S_1 \leftarrow (C_5 \vee C_6)$
$P_{C3} = C_4 \leftarrow True$	$P_{S3} = S_2 \leftarrow C_6$
$P_{C4} = C_5 \leftarrow True$	$P_{S4} = S_3 \leftarrow C_3$
$P_{C5} = C_6 \leftarrow True$	$P_{S5} = S_4 \leftarrow C_2$
$P_{C6} = C_7 \leftarrow S_2 \vee S_4$	

The above is the certificate set and access control strategy defined by unlocking resource R. The certificates the value after whose arrow is true are the public ones which can directly disclose. According to the calculating formula of the effectiveness of certificates mentioned above, the effectiveness of each certificate is as follows, the original effectiveness of each certificate is at the left of following, the effectiveness after disclosing is at the middle, the loss of the effectiveness is at the right. It can be seen that after disclosing, because the privacy loss, the value has a decrease and because of the existence of the certificate C_1, the requirement to this certificate will lead to the failure of negotiation. Thus, the special handing is necessary which is to define the loss as ∞:

$S_1 = 5, C_2 = 5;$	$S_1 = 3, C_2 = 3;$	$S_1 = 2, C_2 = 2;$
$S_2 = 4, C_3 = 5;$	$S_2 = 2, C_3 = 3;$	$S_2 = 2, C_3 = 2;$
$S_3 = 4, C_4 = 2;$	$S_3 = 3, C_4 = 2;$	$S_3 = 1, C_4 = 0;$
$S_4 = 4, C_5 = 2;$	$S_4 = 2, C_5 = 0;$	$S_4 = 2, C_5 = 2;$
$C_1 = 0, C_6 = 2;$	$C_1 = -\infty, C_6 = -1;$	$C_1 = \infty, C_6 = 3;$
$C_7 = 4;$	$C_7 = 3;$	$C_7 = 1;$

When some certificates are of conjunction, the income is the effectiveness of selectable certificates. For example, $C_5 \vee C_6$, if choosing the certificate C_5, the effectiveness is 2 and the loss is 2 When some certificates are of disjunction, the income is the sum of the effectiveness of selectable certificates. For example, $C_5 \wedge C_6$, the effectiveness is 4 and the loss is 5. The income of negotiation strategy defines as follows:

$$U = \sum_{i=1}^{n} U_c(C_i) + \sum_{j=1}^{m} U_s(S_j) + \sum_{i=1}^{n} (\delta(S_i) + \delta(C_i)) \quad (2)$$

Construction of trust game tree: To discuss the construction of trust game tree need to define firstly the algorithm of the action set of participators A (t), according to the access control strategy set at last segment, in order to answer the request of unlocking resource R, the resource provider can divide three actions which are $C_{1(\infty)} \wedge C_{2(2)}$, $\{C_{5(2)}\} \wedge C_{3(2)}$ and $\{C_{6(3)} \wedge C_{7(1)}\}$. In Fig. 4, according to the algorithm of action set, to calculate the game tree is shown:

The algorithm defines as follows:

Input:	The sequence set Π (t) before the game terminates, $(C_1 \wedge C_2)$ $(C_3 \wedge C_5)$ $(C_6 \wedge C_7)$
Output:	A (t')
Step 1:	According to the sequence set Π (t), via the conjunction ∨ as the identifier to split the sequence as Π (t_1), Π (t_2), Π (t_3)
Step 2:	To analyze three sequences one by one using the disjunction ∧ as identifier, such as C_5 mark as {C_5}
Step 3:	To obtain the action sequence set that the resource requestor S can select A (t_1), A (t_2), A (t_3) Definable algorithm of the construction of the game tree is as follows:
Input:	The process of automatic trust negotiation is T = (P, T ,Π, U)
Output:	The game tree Treer Supposing that all the certificate sets of an action sequence from root to the current node t is A (h), also known as historical action set
Step 1:	To find the root node of Treer, marked as S, to press the triple (A (h), A (t), K) into the Stack
Step 2:	If the Stack is non-empty, to take out the top element (A(h), A(t), K). If the mark K is S, go to Step 3; if the mark K is C, go to the Step 4; if the Stack is empty, the algorithm end
Step 3:	To calculate the action set A_S(t') of S: • If it exists the certificate whose effectiveness is in the action set A_S (t'), explaining that there is no such certificate and the negotiation fails, add the final income (-∞,-∞) in node K

- If in the action set A_S (t') there exists some non-public certificates the same as in the historical action set A_S (h'), illustrating that this sequence exists the certificate circle dependency and the negotiation fails, add the final income (-∞,-∞) in the node K
- If the action set is a public certificate, proving that the negotiation succeeds, calculate the terminal income and add the final effectiveness (x,y) in node K
- If not conforming all the situation above, add a node K_{sub} and mark C in node K, merge A_S (t') into A_S (h') and press the triple (A_S (h'), A_S (t'), K_{sub}) into the Stack

Step 4: Calculate the action set A_c (t') of C similar as Step 3 and go to the Step 2 when end

Algorithm of sub game perfect nash equilibrium of trust negotiation: According to the definition, any complete information dynamic finite game exist Nash Equilibrium, to solve sub game perfect Nash Equilibrium uses the backward induction. The essence of the backward induction is a kind of recursive algorithm, roughly describing that deriving from leaf node to the root node, the decision in every step will choose the path of maximum income for itself and finally the formative sequence is the one of sub game perfect Nash Equilibrium, the description of the algorithm is as follows:

```
Input: the extensive game tree: Tree
Output: the sub game perfect Nash Equilibrium of the extensive game tree:
SPE
// calculate the sub game perfect Nash Equilibrium of the game tree
Function CaculteSPE(Tree)
    n = Tree.height; // select the height of trust game tree
    Player = FindRootOfTree(Tree); // select the root of trust game tree
    If ( n > 1)
        SubTreeSet = FindSubTreeSet(Tree); // select all the sub tree
        For each SubTree in SubTreeSet
            // solve the sub game perfect Nash Equilibrium in
        recursion
            ResultSet.insert(CaculteSPE(SubTree));
    Endfor
        // according to the result of recursion to calculate the optimal
        choice of participators
        Return Find Player Best Choice (Player, ResultSet)
    Else
        LeaveSet = FindLeaveSet(Tree);  // obtain the leaf node and
        take out the income
        For each Leave in LeaveSet
            // press the action sequence every leaf node represents into
            the result set
            ResultSet.insert(FindPayoff(Leave));
     Endfor
            // select the optimal result of the sub tree in this floor,
     being sub game perfect Nash Equilibrium
            Return FindPlayerBestChoice(Player, ResultSet);
    Endif
End Function
    // select the optimal income function of appointed participants from
    result set
    Function FindPlayerBestChoice(Player, ResultSet)
        ChoiceSet = FindMaxPayoff(Player, ResultSet);    // select
        a optimal income of a participant
        // select a optimal income of another participant and return
        from former statement
        Return FindMaxPayoff(AnotherPlayer, ChoiceSet);
End Function
```

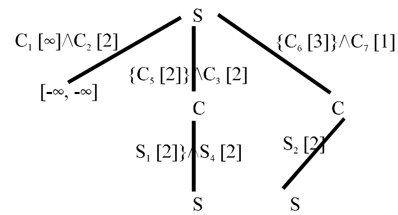

Fig. 4: The example of action set algorithm

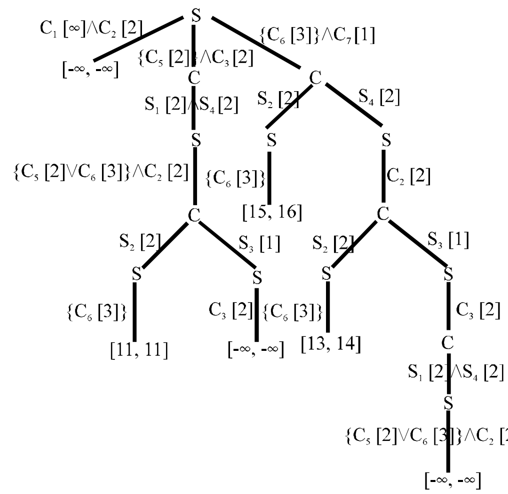

Fig. 5: The example of extensive game tree

Application example: In order to obtain the resource R, it can construct the game tree describing in Fig. 5 as follows. According to the game tree mentioned above, the action sequence cannot the certificate C_1 and cannot build trust and therefore the final income is (-∞,-∞). In addition when the game terminates, the action sequence (($\{C_5\}$,C_3) (S_1,S_4) ($\{C_5,C_6\}$,C_2)(S_3)(C_3)) and (($\{C_6\}$,C_7) (S_4) (C_2)(S_3) (C_5) (S_1,S_4) ($\{C_5, C_6\}$,C_2)) need obtain the certificates C_3 and C_2 which are the same as in former one time. It forms the circle sequence of certificates and cannot construct trust and thus the terminal income is (-∞,-∞). These three sequences can eliminate. There are three action sequence can construct trust:

- The income is ($\{C_5\}$,C_3→S_1,S_4→$\{C_5$,$C_6\}$,C_2→S_2→$\{C_6\}$), the income is (Squicciarini *et al*., 2011, 2010)
- The income is ($\{C_6$,$C_7\}$→S_2→C_6), (Jun-Ichi *et al*., 2010)
- The income is $\{C_6$,$C_7\}$→S_4→C_2→S_2→C_6), (Liu and Lu, 2009; Bonatti *et al*., 2010)

Solve the sub game perfect Nash Equilibrium of the extensive game by using the algorithm mentioned in the former section, obtain the sequence ($\{C_6$,$C_7\}$→S_2→C_6) as optimum which can lead to that both of the game construct trust, expose the minimum number of certificates and decrease the communication burden.

GAME MODEL OF CACHING SEQUENCE

Caching sequence: Trust negotiation process is a complete information static non-zero game (Wang and Yu, 2011). Resource requester and resource provider on the basis of the successful completion of negotiations make it as soon as possible. In other words, maximize the time income (Niyato and Hossain, 2008). The establishment of mutual trust is essentially a strategy to unlock the other side and the certificate process. For example, the resource requester C and resource provider S, C requested to S for resource R, need to provide in accordance with policy set $(R \leftarrow (C_1 \wedge C_2) \vee (C_3 \wedge C_5))$ Unlock R. The C in accordance with the policy set in the request for the certificate $(C_1 \wedge C_2)$ or $(C_3 \wedge C_5)$. But C_2, C_3 are non-public certificate, Requiring in accordance with the policy set of C $(C_2 - (S_2S_3))$ Unlock C_2 and S_2 is non-public certificate, needs C in accordance with the certificate C6 to unlock. Repeat this process both sides, until the completion of this consultation. If R is unlocked, then the negotiation succeeds. In Fig. 6, there is a typical process via constructing the negotiation tree (Squicciarini *et al.*, 2011, 2010) to unlock the resources and certificates.

In Fig. 6, the part of deep color is the revealing sequence of certificates to complete this negotiation. If the strategy sets and certificates in C and S are all unchanged, cache this subsequence and be able to skip the process of negotiation next time to repeat this game. However, storage burden may be larger, especially when the caching sequence (Liu and Lu, 2009) is huge. If C and S change the strategy set, some unnecessary certificates with high sensitiveness (Bonatti *et al.*, 2010) which cause the safety wastage. Thus, dismantle the full caching sequence to many subsequences and each subsequence unlocks a certificate and strategy. Therefore, split this sequence to many sub-sequences which can unlock a certificate and strategy and selectively cache these sub-sequences. When negotiating again, according to needing to use these sub-sequences, package to be complete sequence to unlock the resource. Figure 7 is the sketch map of part-caching sequence.

Theoretically speaking, this method can decrease the storage burden and also decrease the probability of revealing sensitive certificates but decrease time income to some extent and increase the complexity of the system. Via constructing the game model, imitate this process of this repeat game (Jun-Ichi *et al.*, 2010) and discuss the scope of application and strengths and weaknesses of part caching.

Game model of complete caching sequence: Firstly discussing the complete caching game model, the

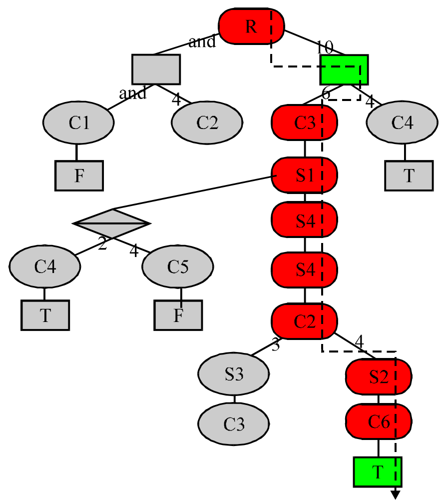

Fig. 6: Negotiation of complete caching sequence

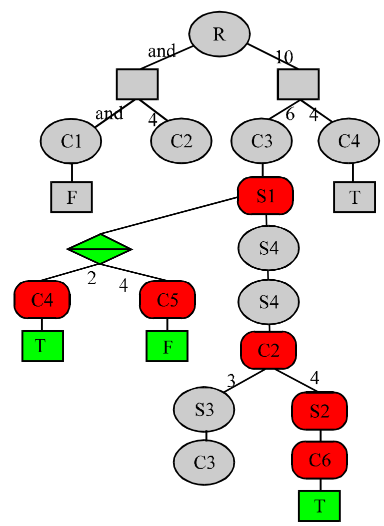

Fig. 7: Negotiation of part caching sequence

resource provider and the resource requester as two game nodes can choose the strategy cache and not cache. Introduce the time income factor: T, the room income factor: s and the safety shortage. In the game model, one side takes the strategy cache and if the other side also takes the strategy cache, both can obtain the income of t. However, caching sequence will bring the safety burden to increase the storage burden. Introduce the room income factor which will bring the storage burden when increasing the time income. To sum up, if one node selects cache, the income formula is s (t-l)-a and if the other node selects not cache because of no storage burden, the income formula is. If the two nodes select cache at the same time, the time income will increase, the income formula is s (2t-l)-a. Table 1 is the two-variable matrix of the complete caching sequence:

Game model of caching sequence adding to ratio factor: On account of problems exposed in the complete caching sequence, trust negotiation model on partial put forward the thought of partial caching sub-sequence whose essence is to introduce the ratio factor η ($0 \le \eta \le 1$) and to simulate caching sequence on partial according to the certain ratio of caching certificate sequence. The introduction of ratio factor caused the room and safety income while losing the time income t. On the one hand when one side chooses to cache, the income formula is ηs (t-l)-a. On the other hand, although choosing to not cache wouldn't consider the influence of space income by the ratio factor, the time income was the same in both two sides because of the specialty of trust negotiation. It would also be influenced by the ratio factor. The income formula was ηst-a. Table 2 shows the two variables income matrix of the caching sequence on partial.

From the income matrix above, it is hard to judge intuitively the influence of adding the ratio factor to the game. To some extent, it depends theoretically on the rate between time and room. The security factor as a constant has the check and balance in the situation of pimping order of magazine but it can ignore when the order of magazine is great. Then, the Nash Equilibrium of improved model is calculated via the game theory and then the influence of ratio factor is discussed via stimulation.

The model mentioned above is a typical mixed strategy game model; it cannot find the Nash Equilibrium of two income matrix by using traditional method in the game theory. According to the definition of mixed strategy game, In the classic game formula with n nodes, e.g., $G=\{S_1,\cdots,S_n;u_1,\cdots,u_n\}$, if to every node I (I = 1, 2, ..., n), P_i*, is the optimal case from the mixed strategy group to the other n-1 nodes which is:

$$u_i(p_i^*,\cdots,p^*_{i-1},p_i^*,p^*_{i+1},\cdots,p^*_n) \ge u_i(p_i^*,\cdots,p^*_{i-1},p_i,p^*_{i+1},\cdots,p^*_n) \quad (3)$$

If P_i could be valid, the mixed strategy group p_i*, ..., p_i*,..., p_n* s Nash equilibrium. To solve the Nash equilibrium for the node game model adding the discount factor we suppose that the possibility of node 1 choosing 'Cache' is and q is for node 2. The income function of both two sides is showed as the following:

$$\begin{cases} u1\,(p1,p2) = pq\,(\eta s\,(t-1)-a) + p\,(1-q)\,(\eta s\,(2t-1)-a) + (1-p)\,(1-q)\,(\eta st-a) \\ \qquad = p\eta st - p\eta s + \eta st - q\eta st - a = qa - pqa \\ u2\,(p1,p2) = pq\,(\eta st-a) + p\,(1-q)\,(\eta s\,(2t-1)-a) + (1-p)\,(1-q)\,(\eta s(t-1)-a) \\ \qquad = p\eta st - q\eta st + \eta st - \eta s + q\eta s - a + qa + pqa \end{cases} \quad (4)$$

Table 1: Game model of complete caching sequence

Resource requester	Resource provider UnCache	Cache
Cache	s (t-1)-a, st-a	s (2t-1)-a, s (2t-1)-a
UnCache	0, 0	st-a, s (t-1)-a

Table 2: Game model of caching sequence adding to ratio factor

Resource requester	Resource provider UnCache	Cache
Cache	ηs (t-1)-a, ηst-a	ηs (2t-1)-a, ηs (2t-1)-a
UnCache	0, 0	ηst-a, ηs (t-1)-a

Then trace the Nash equilibrium with mixed strategy which is the solution of optimal decision:

$$\begin{cases} \max_{p_1 \in P_1} u1\,(p1,p2^*) = \max\,[p\eta st - p\eta s + \eta st - q^*\eta st - a + q^*a - pq^*a] \\ \max_{p_2 \in P_2} u2\,(p1^*,p2) = \max\,[p^*\eta st - q\eta st + \eta st - \eta s + q\eta s - a + qa - p^*qa] \end{cases} \quad (5)$$

Conducting the first derivation of the optimal, we could get the extreme value:

$$\begin{cases} \dfrac{\partial U_1}{\partial p} = \eta st - \eta s - aq^* = 0 \\ \dfrac{\partial U_2}{\partial p} = -\eta st + \eta s + a + p^*a = 0 \end{cases} \quad (6)$$

The result is:

$$\begin{cases} p^* = \dfrac{\eta s\,(t-1)}{a} - 1 \\ q^* = \dfrac{\eta s\,(t-1)}{a} \end{cases} \quad (7)$$

The Nash equilibrium is:

$$\begin{cases} p^*_1 = (\dfrac{\eta s\,(t-1)}{a} - 1, 2 - \dfrac{\eta s\,(t-1)}{a}) \\ p^*_2 = (\dfrac{\eta s\,(t-1)}{a}, 1 - \dfrac{\eta s\,(t-1)}{a}) \end{cases} \quad (8)$$

If selecting Cache, the expected income is showed as the following when P = 1:

$$u_1{}'(p'_1, p_2) = -q\eta st + 2\eta st - \eta s - a \quad (9)$$

If selecting UCache, the expected income is showed as the following when P = 0:

$$u_1{}''(p''_1, p_2) = (a - \eta st)\,q + \eta st - a \quad (10)$$

In the end, P1 is the optimal decision for the node strategy. No matter whether selecting Cache or UCache, it wouldn't influence the final result of the game:

$$u_1(\text{"Cache"}, p_2) = u_1(\text{"UnCache"}, p_2) \tag{11}$$

then:

$$q = \frac{\eta s\,(t-1)}{a} \tag{12}$$

Therefore, when the probability for node 2 selecting Cache is:

$$q < \frac{\eta s\,(t-1)}{a}$$

node 1 selects Cache Otherwise, if:

$$q > \frac{\eta s\,(t-1)}{a}$$

node 1 selects 'UCache' When:

$$q = \frac{\eta s\,(t-1)}{a}$$

node 1 selects 'Cache' or 'UCache' randomly.

Discussion for the effectiveness of ratio factor: In case that in a certain moment, there is a game between the resource provider and requester, according to the result above:

$$0 \le p = \frac{\eta s\,(t-1)}{a} \le 1$$

and

$$0 \le 1 - p = 1 - \frac{\eta s\,(t-1)}{a} \le 1$$

we could found that the range of ratio factor is:

$$0 \le \eta \le \frac{a}{s\,(t-1)}$$

from that it is not difficult for us to make out that the more the s (t-l) is, the less the ratio factor is namely, caching sequence is the less the better. Therefore, we could raise the ratio factor formula as:

$$\eta = \frac{a}{st}$$

According to the formula above when the safety loss approximate to the product of time and space income st, the partial caching ratio is approximately 1 which mean the partial caching sequence valid. Otherwise, considering the practice of integer partite, the probable value for ratio factor is 75, 50 or 30%. So, the possible value of ratio factor would be less than 50% when is very large which would deplete the time income of trust negotiation greatly. In a word, by solving the Nash equilibrium of mixed game and analyzing the reality, we find that ratio factor can't optimize the income of trust negotiation obviously. Especially, it would lead to the loss of time income. Therefore, it's tiny effective to improve the integral income by introducing the partial caching sequence in the research of trust negotiation framework.

SIMULATION RESULTS AND ANALYSIS

Gambit is a kind of simple and useful software for game simulation which could conduct simple and expanding game simulation. Mixed strategic game belongs to simple game. Supposing that resource requester and provider as Client and Service, respectively, among which Cache and UCache represent caching strategy and un-caching strategy, respectively. And more than once games are repeated. We define time profit, safety loss and space factor as 5, 3 and 1.5, respectively, conducting the simulation as the following. Figure 8 is the game stimulating picture of not adding the ratio factor. Figure 9 are the game simulating pictures of adding the ratio factor, in which η = 1.0 of a), η = 0.75 of b), η = 0.50 of c), η = 0.25 of d).

According to the simulating pictures above when the abscissa Lambda has different values, to compared with the convergence of each ratio factor. Figure 10 is the picture of the curve converges.

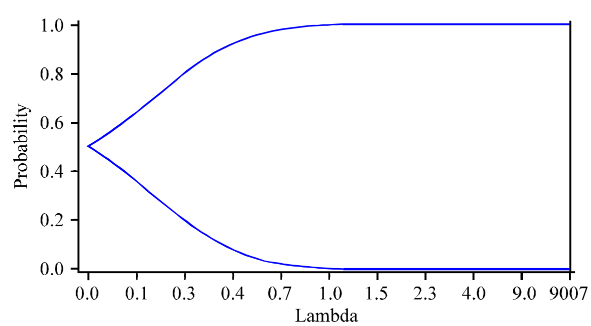

Fig. 8: Simulation result of not adding the ratio factor

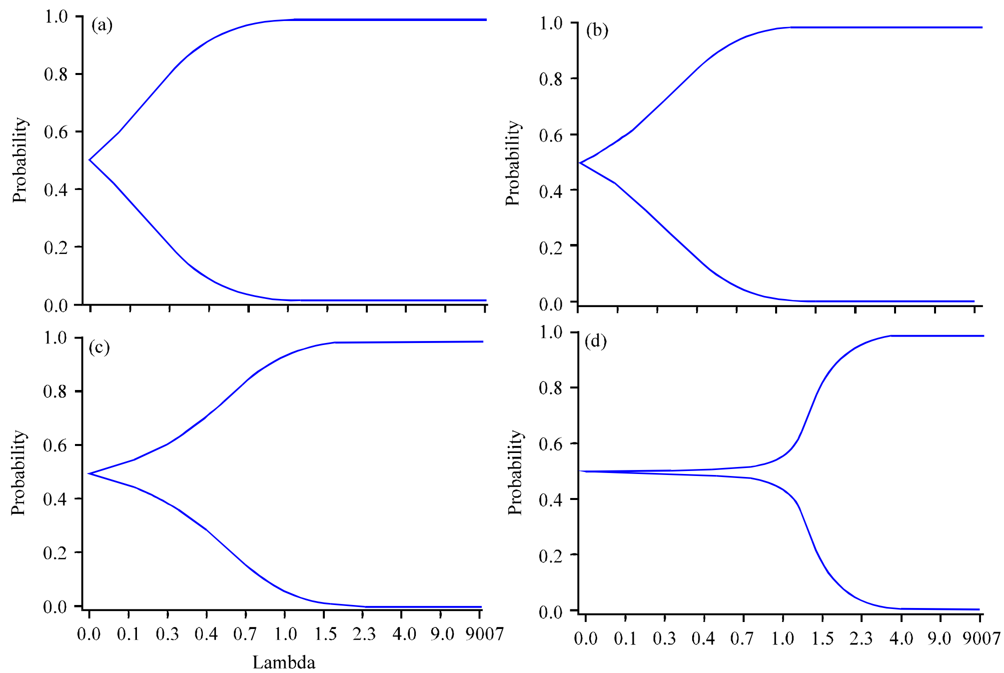

Fig. 9 (a-d): Simulation result of adding the ratio factor; (a, b) η= 1.0, (c) η = 0.5 and (d) η = 0.25

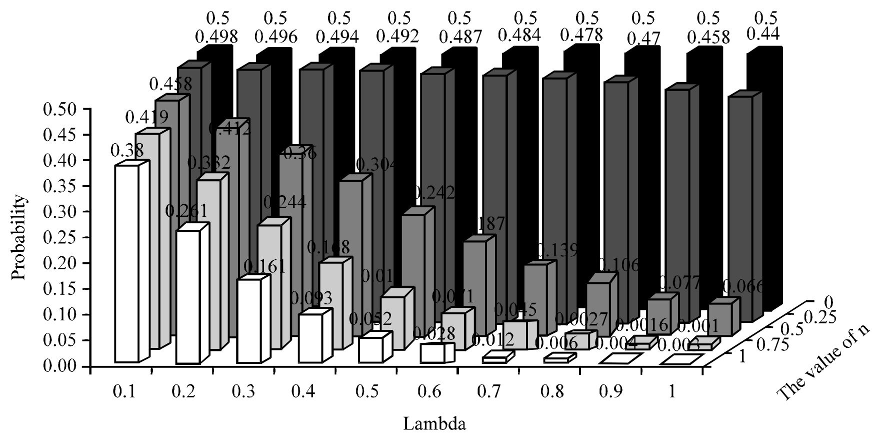

Fig. 10: Comparison of curve converges

CONCLUSIONS

According to the specialty of trust negotiation, present study introduces the theory of complete information active game and on the purpose of improve the structure of trust negotiation, puts forward a kind of strategy of trust negotiation, by using both of negotiation as decision nodes and the exposable certificate set as the actions and constructing the game tree at a time. In addition, the distinction between complete caching sequence and caching sequence on partial is discussed. Via the cache strategy of mixed strategy model simulating the trust negotiation, importing the ratio factor to distinguish the situations of complete caching and caching on partial, then calculating the Nash equilibrium and mathematical analysis. Finally, adjust the ratio factor and simulating, it confirms that in the negotiation structure, caching sequence on partial has no effect to increase the total income. With the combination between the game theory and P2P trust negotiation, this strategy reaches the equilibrium between effectiveness and security and provide a kind of thought for improving the existing structure of trust negotiation. Furthermore, discussing the feasibility of some improving projects has some positive effect to the research of trust negotiation strategy set and certificate sequence.

At present, the elementary discussion for this strategy has been done. The influence of repetition game to negotiation result and some complicated potential safety hazard, such as access control strategy and certificate forged, has not discussed. In the future, the encipherment of the certificate set and its evolution will be lucubrated.

ACKNOWLEDGMENT

This study was supported by Natural Science Foundation of Hubei Province of China (2010CDA011), Foundation of Wuhan Twilight Plan Project (201050231084) and Educational Commission of Hubei Province of China (D20111409). We would like to thank the reviewers and editors for their detailed and valuable comments.

REFERENCES

Ardagna, C.A., M. Cremonini, S. De Capitani di Vimercati and P. Samarati, 2008. A privacy-aware access control system. J. Comput. Security, 16: 369-397.

Bonatti, P.A., J.L. De Coi, D. Olmedilla and L. Sauro, 2010. A rule-based trust negotiation system. IEEE Trans. Knowledge Data Eng., 22: 1507-1520.

Chou, S.Y., S.W. Lin and C.C. Li, 2008. Dynamic parking negotiation and guidance using an agent-based platform. Exp. Syst. Applic., 35: 805-817.

Jun-Ichi, I., O. Makoto and Y. Chikara, 2010. Partial tax coordination in a repeated game setting. CESifo Working Paper Series, No. 3127.

Li, J., D. Zhang, J. Huai and J. Xu, 2009. Context-aware trust negotiation in peer-to-peer service collaborations. Peer-to-Peer Networking Applic., 2: 164-177.

Liu, B. and H. Lu, 2009. A peer-to-peer frame work for accelerating trust establishment. Int. Conf. Multimedia Inform. Networking Security Mines, 1: 135-139.

Lu, H. and B. Liu, 2009. DFANS: A highly efficient strategy for automated trust negotiation. Comput. Security, 28: 557-565.

Niyato, D. and E. Hossain, 2008. Modeling user churning behavior in wireless networks using evolutionary game theory. Proceedings of the Wireless Communications and Networking Conference, March 31-April 3, IEEE., pp: 2793-2797.

Skogsrud, H., H.R. Motahari-Nezhad, B. Benatallah and F. Casati, 2009. Modeling trust negotiation for web services. J. Comput., 42: 54-61.

Squicciarini, A.C., F. Paci, E. Bertino, A. Trombetta and S. Braghin, 2010. Group-based negotiations in P2P systems. IEEE Trans. Parallel Distrib. Syst., 21: 1473-1486.

Squicciarini, A.C., F. Paci and E. Bertino, 2011. Trust establishment in the formation of virtual organizations. Comput. Standards Interfaces, 33: 13-23.

Traum, D., S.C. Marsella, J. Gratch, J. Lee and A. Hartholt, 2008. Multi-party, multi-issue, multi-strategy negotiation for multi-modal virtual agents. Lecture Notes Comput. Sci., 5208: 117-130.

Wang, G. and Z. Yu, 2011. A pontryagin's maximum principle for non-zero sum differential games of BSDEs with applications. IEEE Trans. Automatic Control, 55: 1742-1747.

Zhang, C.C. and M. Winslett, 2008. Distributed authorization by multiparty trust negotiation. Lecture Notes Comput. Sci., 5283: 282-299.

Zhang, Q., X. Wang and Z. Gong, 2008. A role-based automated trust negotiation and authentication design in Mobile Ad Hoc Networks. Proc. Int. Conf. Comput. Sci. Software Eng., 3: 688-691.