WUJNS
Wuhan University Journal of Natural Sciences

Vol. 11 No. 6 2006 1462-1466

Article ID:1007-1202(2006)06-1462-05

Time-Based Dynamic Trust Model Using Ant Colony Algorithm

□ TANG Zhuo, LU Zhengding†, LI Kai
College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, Hubei, China

Abstract: The trust in distributed environment is uncertain, which is variation for various factors. This paper introduces TDTM, a model for time-based dynamic trust. Every entity in the distribute environment is endowed with a trust-vector, which figures the trust intensity between this entity and the others. The trust intensity is dynamic due to the time and the inter-operation between two entities, a method is proposed to quantify this change based on the mind of ant colony algorithm and then an algorithm for the transfer of trust relation is also proposed. Furthermore, this paper analyses the influence to the trust intensity among all entities that is aroused by the change of trust intensity between the two entities, and presents an algorithm to resolve the problem. Finally, we show the process of the trusts' change that is aroused by the time's lapse and the inter-operation through an instance.

Key words: dynamic trust; ant colony algorithm; Inter-operation; time-based
CLC number: TP 305

Received date: 2006-05-12
Foundation item: Supported by the National Natural Science Foundation of China (60403027), Natural Science Foundation of Hubei Province (2005ABA258) and Open Foundation of State Key Laboratory of Software Engineering (SKLSE05-07)
Biography: TANG Zhuo (1981-), male, Ph. D. candidate, research direction: security for distributed system. E-mail: hust_tz@126.com
† To whom correspondence should be addressed. E-mail: zdlu4409@public.wh.hb.cn

0 Introduction

With the development of Internet-based services and applications proceeding at unprecedented speed, a mass application systems turn to distributing from centralizing. A practical problem is to implement the inter-operation between the different information sources, how to construct the trust relationship between entities in the complex distributed heterogeneous system that is a challenge for the researchers. Since Marsh[1] has imported trust research into computer area, trust mechanism has received more and more attention because of its flexibleness and extensibility. Many trust models[2] are proposed in a lot of areas such as distributed network, P2P computing, ubiquitous computing[3] and autonomous network[4]. In the traditional model of the trust relationship, trust was usually defined as a Boolean variable[5], that is to say, in the session of both trust entities, one trust another entirely, or absolutely not, there would never be a middle course. For instance, the entity *A* trusts entity *B*, but it is hard to tell how much they trust each other. Therefore, we have to quantify their trust intensity, so that we can define the meaning mentioned above. It is claimed in this paper that the trust intensity between entities is not fixed. In fact, the trust intensity is varied with the inter-operation and time lapse.

We propose a new dynamic trust relationship model that the trust intensity is the function of the time and the inter-operation events. In the model, the trust intensity will be described by the trust-degree, base on the mind of the ant colony algorithm. Ant colony algorithm is a kind of simulated evolution algorithm based on the research on the ant colony's behaviors in the nature. Although single ant's behavior may be too simple, but colony of these simple ants may behave in a complex way and complete intricate tasks. Moreover, ants may responds to the change of environment. For example, when encountering an obstacle in their route, ants can find optimal detour route quickly. People find that ant communicates with each other through so-called pheromone matter to collaborate to cover tasks.

As described in ant colony algorithm, when an ant passed a way, it has leaved some pheromone in the way. When more ants pass the way, the pheromone's amount in the way is stronger and stronger. Adversely, if there is no ants pass the way, the pheromone's amount in the way will reduce after a while. Colorni and Dorigo *et al*[6,7] proposed the method to formalize the change of the pheromone. And we think that the trust intensity is quite similar to the pheromone. For example, the fact that the trust intensity will increase when two entities have an inter-operation is similar to the increase of the pheromone in a way. Furthermore, as mentioned above, the condition of the decrease of the pheromone in a way is similar to that of the reduction of the trust intensity between entities.

Based on the mentioned above, we propose the trust-pheromone similarly. Every inter-operation between two entities in the system can increase their trust-pheromone, and the trust-degree will decrease with time passing by. If the two entities have not communicated with each other for a lone time, the trust-pheromone will decrease correspondingly.

1 TDTM Model

1.1 Entity's Trust Relation

Definition 1 Trust-degree describes entity's trust level, it can be represented as $p_{ij}(t)$, its value ranging between 0 and 1, $p_{ij}(t) = 0$ expresses that the entity i and entity j don't trust each other at time t, and $p_{ij}(t) = 1$ expresses that the entity i and entity j trust each other completely at time t, reversely.

Definition 2 Trust-vector notes the trust-degree between this entity and others suppose there are n entities in the distributed environment, and then every entity has a local trust vector:

$$p_i(t) = (p_{i_1}(t),\ p_{i_2}(t), \cdots, p_{i_n}(t)) \quad (1)$$

This vector is the function for inter-operation event and time. If entity i and entity j have inter-operation at time t, then the trust-degree between the two entities will increase, it as follows:

$$p_{i_i}(t) > p_{i_i}(t-1) \quad (2)$$

After a lapse of interval, the trust-degree between any two entities will decrease tardily, if they have not any inter-operation in a quite long time, the trust degree will change to zero, and that is to say the two entities will have not any trust each other.

When system initializes, every entities' local trust vector will be confirmed, it is described as $p_i(0)$. The policy we adopted is as follows:

If the whole distributed system is under a CA that is a hierarchy, and when the two entities under the same CA, the trust degree between them is 1, reversely it is zero.

If the CA configuration is a reticulation, and when there is a trust path between the two entities, the trust degree between them is 1, reversely it is zero.

1.2 Description of Dynamic Trust

In order to make the increase and reduce of the dynamic trust between two entities numeric, we introduce trust-pheromone.

Definition 3 Trust-pheromone describes the base factor between two entities, let $\tau_{ij}(t)$ be the trust-pheromone between entities i, j at time t. Under the initial condition, we can set the value of the trust pheromone between entities by practice case, possibly, we let the trust-pheromone equal at the beginning, let $\tau_{ij}(0) = C$ (C is a constant). If the trust-degree between entities $p_{ii}(0) = 0$, then $\tau_{ij}(0) = 0$.

And we assume that η_{ij} is the trust-transparence[8], this paper definite that it is the heuristic information that a entity trust in another, we set $\eta_{ij} = 1/d_{ij}$, d_{ij} is the distance between entity i and entity j, α is the trust information's weightiness in the path from i to j, β is the weight of heuristic information (whose initial value is base the actual case).

According the pheromone's expressions in ant colony algorithm [9,10], at time t, the trust-degree between entities is defined as:

$$p_{ij}(t)\,[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta} / [\sum_{i,j}[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}],\ i \neq j \quad (3)$$

If there is no inter-operation between entities, the strength of trust-pheromone between two entities will decrease tardily, let $1-\rho$ be the diluteness of trust-pheromone intensity, and $\sigma\tau_{ij}$ be the additional trust-pheromone intensity at each inter-operation between entities.

$$\tau_{ij}(t+n) = \rho\tau_{ij}(t) + \sigma\tau_{ij} \quad (4)$$

Where:

$$\sigma_{ij} = \begin{cases} \dfrac{1}{\dfrac{1}{1 - \tau_{ij}(t)} + 1}, \\ \quad \text{if entities } i, j \text{ interoperation at time } t \\ 0, \text{otherwise} \end{cases} \quad (5)$$

1.3 Transfer of Trust Relation

According to determinate theory, we can define the transfer of entities' trust-pheromone as:

$$\tau_{i,i+2}(t) = \tau_{i,i+1}(t) \times \tau_{i+1,i+2}(t) \quad (6)$$

And if the trust-pheromone which is received through trust transfer larger than the entities' directs trust-pheromone, for instance, at some time t, if $\tau_{i,i+1}(t) \times \tau_{i+1,i+2}(t) > \tau_{i,i+2}(t)$, then we change the value of the entities' direct trust-pheromone:

$$\tau_{i,i+1}(t) \times \tau_{i+1,i+2}(t).$$

1.4 Calculation of Trust-Pheromone

We can describe the relationship of the entities' trust as an undirected graph; each entity is a vertex in the graph, the edges between two vertexes represents that there is trust relationship between the two entities, and the weight of the edge represents the trust-pheromone between the entities, so we can use a matrix to describe the graph. Arbitrary two entities' trust-pheromone can be received by two means: the first is the weight of the edge between two vertexes, which notes as v; the other is the indirect weight of the simple path which from one entity to another. We assume there are n paths exist between entity i and j, the length of the simple path is l_k, $k \in [1, n]$, the weight of an edge in the path is v_{ij}. Based the transfer relationship of trust-pheromone above, we can get that each simple path's weight between two entities is:

$$\prod_{1}^{k} v_{i,j}, \ k \in [1, n]$$

Finally, the two vertexes' practical trust-pheromone is:

$$\max(v, \prod_{1}^{I_k} v_{i,j}) \ k \in [1, n] \quad (7)$$

We can set a value R for the trust-degree's delimitation, which show whether the entities can validate their certificate. If $p_{ij}(t) > R$, that denotes the entity i, j have enough trust-degree, so they can validate their certificate each other, not otherwise.

2 The Update of Trust-Pheromone

As mentioned above, the trust-pheromone are inconstant because the time lapse and the inter-operation between entities. So, this reflect in the undirected graph is that the change of weight between two entities i, j will arose the change of weight between two specifically entities, which have a path from one to another pass entity i and j. So, we must adjust the weights of the whole edges adapt to the variety.

2.1 Algorithm Description

This algorithm describes the update of trust-degree when the trust-pheromones between entities v and s change at some time. When these changes happen, we have to modify trust-pheromones between every two entities while there is a path passes the entities v and s from the one entity to the other.

An update method based DFS algorithm is proposed in this paper. First of all, deep first transverse from the node v_1, find all path from v_1 to $v_2: p_i(\langle v_1, v_2\rangle \notin p_i)$. Obviously, if adding the edge $\langle v_1, v_2\rangle$ to p_i, we can get a circle S which contain the edge $\langle v_1, v_2\rangle$, S is described as $S: v_1, s_i, s_n, v_2, v_1$. Set variable W as the product of all edges' weights, V_{s_i,s_j} as the weight of two arbitrary adjoining nodes s_i, s_j, the actual weight between two nodes is:

$$P_{s_i,s_j}(t) = \max(V_{s_i,s_i}, W / V_{s_i,s_i}) \quad (8)$$

The update algorithm is as follows:

i) Work out the aggregate H contain all circles which pass the edge $\langle v, s\rangle$, $H = (S_1, S_2, \cdots, S_h)$, h is the number of the circles.

ii) For each circle $S_i \langle v_1, v_2\rangle (i \in [1, h])$, according formula (8), update every edge's weight in it.

The particular steps are as algorithm 1 shows:

Algorithm 1 The algorithm of updating trust-pheromone.

① Find all simple paths from v to s according DFS algorithm;

② Let each path be a circle through adding a edge $\langle v, s\rangle$;

③ Calculate variable W as the product of all edges' weights in a circle in turn;

④ For each two direct adjoin entities s_i, s_i in each circle, calculate the trust-degree from trust transfer through the other edges in the circle: $\tau_{s_i,s_i}(t) = W / V_{s_i,s_i}$;

⑤ If $\tau_{s_i,s_i}(t)$ is larger than the direct trust-degree for the adjoin entities, then set $V_{s_i,s_i} = \tau_{s_i,s_i}(t)$, otherwise not change.

⑥ Calculate trust-score between each pair entities according this formula:

$$p_{vs}(t) = [\tau_{vs}(t)]^{\alpha}[\eta_{vs}]^{\beta} / [\sum_{v,s}[\tau_{vs}(t)]^{\alpha}[\eta_{vs}]^{\beta}]$$

2.2 Complexity Analysis

The description above manifest that the algorithm's complexity is materialized in the update of the trust-degree, because this paper adopts local trust-vector to describe the trust-degree between adjoin entities, so we can figure the all trust-vectors using an adjoin table. We can get the conclusion that the complexity of the algorithm is $O(n^2)$ through the analysis to the steps of the algorithm and the DFS algorithm.

Proposition 1 The complexity of the algorithm for the update of the trust-pheromone is $O(n^2)$.

Proof Based on the mention above, when calculating all paths between two nodes, the complexity of finding one entity's adjoin nodes is $O(e)$, e is the number of the edges. Thus, the complexity when calculating all paths between the two entities is $O(n+e)$, n is the number of the system's entities. Since each trust-vector's dimension is n, actually, e is $O(n^2)$ quantitatively, so the complexity of step ①is $O(n^2)$, in step ③,for each circle, calculating the product of all edges' weights, the complexity is $O(n^2)$, other steps' complexities are $O(n)$. Thus it can be seen that the complexity of the algorithm is $O(n^2)$.

3 Examples

We assume that there are four entities in a distributed environment, the value of various parameters is set as: $\alpha = 1.0$, $\beta = 5.0$, $\rho = 0.9$, $n = 4$, $\tau_{ij}(0) = 100$, the initial undirected graph is as Fig. 1.

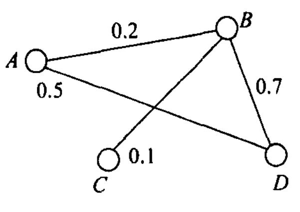

Fig. 1 The initial undirected graph

From this graph, we can get the initial adjoin matrix

$$M_1 = \begin{bmatrix} 1 & 0.2 & 0 & 0.5 \\ 0.2 & 1 & 0.1 & 0.7 \\ 0 & 0.1 & 1 & 0 \\ 0.5 & 0.7 & 0 & 1 \end{bmatrix}$$

According the transfer of trust relation algorithm, the initial undirected graph has to be adjusted. For example, there are two simple paths between the entities A and C: ABC, ADBC. We can calculate each simple path's weight according formula (6), then the trust-degree between A and C must to be updated as: max(0.02, 0.035) = 0.035. And we can adjust the other edges' weight similarly. Generally, we can use the adjoin matrix M_2 to depict this state.

$$M_2 = \begin{bmatrix} 1 & 0.2 & 0.035 & 0.5 \\ 0.2 & 1 & 0.1 & 0.7 \\ 0.035 & 0.1 & 1 & 0.07 \\ 0.5 & 0.7 & 0.07 & 1 \end{bmatrix}$$

The element in the row i and column j denotes the trust-pheromone $\tau_{ij}(0)$ between the entities i and j.

As this paper stated above, we can analyze that the trust-pheromone various follow the change of time and the inter-operation event in this instance. If there are no inter-operation, the trust-pheromone between arbitrary entities will decrease after a time unit t, due to the parameter $\rho = 0.9$. This situation can be formalized as the matrix

$$M_3 = [\tau_{ij}(t)] = \begin{bmatrix} 1 & 0.18 & 0.0315 & 0.45 \\ 0.18 & 1 & 0.09 & 0.63 \\ 0.0315 & 0.09 & 1 & 0.063 \\ 0.45 & 0.63 & 0.063 & 1 \end{bmatrix}$$

It doesn't need to adjust the trust-degree due to the transfer of trust.

If an inter-operation is found at time $2t$ between entities A and B, according formulas (4) and (5),

$$\begin{aligned} \tau_{AB}(2t) &= \rho\tau_{AB}(t) + \sigma\tau_{AB} \\ &= 0.9 \times 0.18 + \frac{1}{\frac{1}{1-0.18} + 1} \\ &= 0.162 + 0.45 = 0.612 \end{aligned}$$

Now we can found all circles which have an edge pass A, B: ACBA, ACDBA, ADBA, according the algorithm of the trust-pheromone's update, the updated matrix of trust-pheromone is:

$$M_4 = [\tau_{ij}(2t)] = \begin{bmatrix} 1 & 0.612 & 0.055 & 0.45 \\ 0.612 & 1 & 0.09 & 0.63 \\ 0.055 & 0.09 & 1 & 0.063 \\ 0.45 & 0.63 & 0.063 & 1 \end{bmatrix}$$

Finally, according formula (3), we can get the trust-degree matrix at time $2t$:

$$M_5 = [p_{ij}(2t)] = \begin{bmatrix} 1 & 0.322 & 0.029 & 0.237 \\ 0.322 & 1 & 0.047 & 0.332 \\ 0.029 & 0.047 & 1 & 0.033 \\ 0.237 & 0.332 & 0.033 & 1 \end{bmatrix}$$

The matrix M_5 shows the trust relation among the four entities. For example, the $p_{12}(2t) = 0.322$, it represent that the trust degree between entities A and B is 0.322 at time $2t$. Obviously, the row vectors are local trust-vectors of the entities, the first row vector is (1, 0.322, 0.029, 0.237), that is to say the trust degree between entity A and others are 0.322, 0.029 and 0.237 respectively.

4 Conclusion

The uncertainty and the heterogeneity of distributed systems are the ultimate challenges for the found of the trust among the entities. This paper presents a model for time-based dynamic trust based the mind of ant colony algorithm. The model indicates that the trust is the function of the time and the inter-operation event. We can quantify the real-time trust relationship through this model. And the model can adjust the global trust-pheromone in time for the change of the local trust-pheromone. Based on the results presented in the previous section, this paper shows the progress of the trust between entities through an instance at last.

We believe that this model can be viewed as a basis for several extensions. For example, this paper supposes that all inter-operations are successful, and it doesn't refer the case that the inter-operation is failed among the entities. We plan to extend TDTM and make it include the analysis of the aborted inter-operation. Another important future work is the applications of the model in trust manage.

References

[1] Marsh S P. *Formalising Trust as a Computational Concept* [D]. Stirling: University of Stirling, 1994.

[2] Beth T, Borcherding M, Klein B. Valuation of Trust in Open Networks[C]// *ESORICS* 94. Brishton, December 1994.

[3] Yu Bin, Munindar SirIsh P. An Evidential Model of Distributed Reputation Management[C]// *Proceedings of First International Joint Conference on Autonomous Entities and Multi-Entity*. Systems, Morgan, July 2002.

[4] Aberer K, Despotovic Z. Managing Trust in a Peer-2-Peer Information System [C]// *Proceedings of the l0th International Conference on Information and Knowledge Management*. New York, October 2001.

[5] Abdul Rahman A, Hailes S. A Distributed Trust Model [C]// *New Security Paradigms Workshop*. New York: ACM Press, 1997: 51-68.

[6] Colorni A, Dorigo M, Maniezz O V. Distributed Optimization by Ant Colonies[C]// *Proceeding of The First European Conference Artificial Life*. Paris: Elsevier Publishing, 1991: 134-142.

[7] Dorigo M, Maniezzo V, Colorni A. The Ant System; Optimization by a Colony of Cooperating Agents [J]. *IEEE Transaction on System*, 1996, **26**(1): 1-26.

[8] Dorigo M, Gambarde L. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem[J]. *IEEE Transaction on Evolutionary Computing*, 1997, **1**(1): 53-56.

[9] Colorni A, Dorigo M, Maniezz O V. An Investigation of Some Properties of an "Ant Algorithm" [C]// *Proceeding of the Parallel Problem Solving from Nature Conference*, Brussels: Elsevier Publishing, 1992: 509-520.

[10] Dorigo M, Cargo G D, Gambardella L M. Ant Algorithm for Distributed Discrete Optimization [J]. *Artificial Life*, 1999, **5**(2): 137-172.

□