International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:3, 2007

Fuzzy Trust for Peer-to-Peer Based Systems

Farag Azzedin, Ahmad Ridha, and Ali Rizvi

***Abstract*—** Trust management is one of the drawbacks in Peer-to-Peer (P2P) system. Lack of centralized control makes it difficult to control the behavior of the peers. Reputation system is one approach to provide trust assessment in P2P system. In this paper, we use fuzzy logic to model trust in a P2P environment. Our trust model combines first-hand (direct experience) and second-hand (reputation) information to allow peers to represent and reason with uncertainty regarding other peers' trustworthiness. Fuzzy logic can help in handling the imprecise nature and uncertainty of trust. Linguistic labels are used to enable peers assign a trust level intuitively. Our fuzzy trust model is flexible such that inference rules are used to weight first-hand and second-hand accordingly.

***Keywords*—** P2P Systems; Trust, Reputation, Fuzzy Logic.

I. Introduction

THE *peer-to-peer* (P2P) computing is one of the technologies that is having a significant impact on the way Internet-scale systems are built. It is well established for applications such as file sharing and parallel distributed computation. The popularity of P2P computing has prompted the research community to examine several aspects of it. One aspect is to extend P2P computing to host a wider variety of applications.

One issue in P2P systems is trust [5], [3], [6]. The manifestation of trust as a crucial issue can be understood by closely looking at traditional P2P applications. In file sharing and parallel computations, a massive redundancy approach is followed. In this approach, the objective is to make the hosted service (in the case of file sharing the access to files) immune to individual resource failures or misbehavior. While this approach yields robust service for applications such as file sharing, it is not suitable for sensitive applications such as hosting databases or storing medical images. One of the causes of this situation is the massive redundancy approach itself. Because of massive redundancy, a peer by itself has very little value to the P2P system in which it is a member. As a result, the peer has little incentive to contribute towards the overall goal of P2P system and usually ends up pursuing its own agenda. This has been observed in real P2P systems as the *free riding* phenomena [8] and also philosophically referred to as the "tragedy of the commons" [4].

Manuscript received March 27, 2007. This work was supported in part by the Deanship of Scientific Research (DSR) at King Fahd University of Petroleum and Minerals (KFUPM) under Grant JFRG-2006-16.

Dr. Farag Azzedin is an Assisstant Professor with KFUPM, P.O. Box 1290, Dhahran, Saudi Arabia. Dr. Farag Azzedin is the corresponding author. Phone: 03-860-3431; fax: 03-860-2174; e-mail: fazzedin@kfupm.edu.sa.

Ahmad Ridha is a Graduate Student with the Department of Information and Computer Science at KFUPM; e-mail:aridha@kfupm.edu.sa.

Ali Rizvi is a Graduate Student with the Department of Information and Computer Science at KFUPM; e-mail:alirizvi@kfupm.edu.sa.

The trust notion is vague [10], [6] and there exists a gray area in expressing the trustworthiness of a peer. This prompted recent researchers [2], [6], [10] to model trust using fuzzy logic. Fuzzy logic uses qualitative terms and linguistic labels to represent trust as a fuzzy concept, where membership functions describe to what degree a peer can be labeled as trustworthy or untrustworthy. Fuzzy logic provides rules for reasoning with fuzzy measures of this type.

In modeling trust, concepts such as trustworthy, honesty, and accuracy are defined and quantified. Since these linguistic labels are fuzzy, we can apply fuzzy logic to handle the uncertainty and the imprecision in any trust model [32].

In previous work, we presented a trust model that manages and models the trust relationships among different peers in a P2P setting. This paper builds on our previous work and contributes the following: (a) we enhanced the concept of honesty and its usage by differentiating between consistency and honesty, (b) we utilized the decay function by including time stamp and transaction frequency, and (c) we used fuzzy logic to model trust representation, trust aggregation, and trust evolution.

In Section II, we define and describe our enhanced trust model. We present our fuzzy trust model in Section III and discuss the current literature for modeling trust using fuzzy logic in Section IV. Section V concludes the paper.

II. Trust Model

A. Fundamental Trust Concepts

Behavior trust is quantified by a dynamic parameter called *trust level* (TL) that ranges from *very untrustworthy* to *very trustworthy*, and which is represented by a numeric range [1..5] The TL is computed based on past experiences for a specific context. For example, based on trust, peer y might allow peer x to use its resources to store data files and not executable files.

Our trust model assumes that each peer x maintains a *set of recommenders* (R_x) and a *set of trusted allies* (T_x). Peer x completely trusts its trusted allies that are chosen based on off-line trust relationships. Trusted allies are used by a peer to determine the honesty of its recommenders. In general, trusted allies of a peer x do not have sufficient knowledge to provide recommendations themselves. The recommenders of x are maintained in a *recommender trust table* (RTT_x), where a two-tuple (honesty, accuracy) is associated with each entry. Similarly, x maintains another table called *direct trust table* (DTT_x) for tracking transactions x had with other peers.

B. Computing Honesty and Accuracy

Peer x randomly chooses its recommender z and initializes it as follows. First, z is considered to have maximum accuracy

regardless of the target peer. That is, $A_x(z,t,c) = 1$ meaning that x considers z to have maximum accuracy for context c at time t. Second, since z is a new recommender and gave no recommendations so far, z's *recommendation error* as observed by x for context c at time t is set to zero (i.e., $\Psi_{RE_x}(z,t,c) = 0$). Third, z is considered consistent regardless of the target peer. That is, $C_x(z,t,c) = 1$ meaning that x considers z consistent for context c at time t. Peer x's objective is keep only honest and accurate recommenders in R_x.

Let the consistency of recommender z as observed by x in giving recommendation regarding y for context c at time t be denoted by $C_x(z,y,t,c)$. Let $RE_k(z,y,t,c)$ denote the recommendation for peer y given by z to peer k for context c and time t, where $k \in T_x$. The recommendation $RE_k(z,y,t,c)$ is given from DTT_z and that is what z believes in as the reputation of y. Let $TL_{min}(x,z,y,t,c) = \min_{k \in T_x}\{RE_k(z,y,t,c)\}$ and $TL_{max}(x,z,y,t,c) = \max_{k \in T_x}\{RE_k(z,y,t,c)\}$. Let $\Delta_{RE_x}(z,y,t,c)$ denote the difference and be given by:

$$\Delta_{RE_x}(z,y,t,c) = TL_{max}(x,z,y,t,c) - TL_{min}(x,z,y,t,c) \quad (1)$$

The value of $\Delta_{RE_x}(z,y,t,c)$ will be less than a small value ϵ_{RE} if recommender z is consistent. Consequently, $C_x(z,y,t,c)$ is computed as follows: (It should be noted that after $C_x(z,y,t,c)$ is computed, it will be used to update z's overall consistency regardless of the target peer (i.e., $C_x(z,t,c)$). Please, see Section II-D).

$$C_x(z,y,t,c) = \begin{cases} 0 & \text{if } \Delta_{RE_x}(z,y,t,c) > \epsilon_{RE} \\ 1 & \text{otherwise} \end{cases}$$

If $C_x(z,y,t,c) = 0$, then z is dishonest and will be filtered out and prevented from influencing the recommendation network. If $C_x(z,y,t,c) = 1$, then z is consistent but may be dishonest. Another filter called the accuracy measure is used by our trust model to capture these consistent but dishonest recommenders and adjust their recommendations before using them to compute the reputation of y.

Seeking the reputation of y, x will ask z for recommendation regarding y for c at time t. Once z sends its recommendation (i.e., $RE_x(z,y,t,c)$) and before x can use $RE_x(z,y,t,c)$ to calculate the reputation of y, $RE_x(z,y,t,c)$ must be adjusted to reflect recommender z's accuracy. To achieve this objective, a *shift function* (S) is introduced that uses the overall accuracy $A_x(z,t,c)$ (i.e., z's accuracy regardless of the target peer) to correct $RE_x(z,y,t,c)$. The shift function is defined as:

$$S(A(x,z,t,c), RE_x(z,y,t,c)) = \begin{cases} RE_x(z,y,t,c) + 4(1 - A(x,z,t,c)) & \text{if } \Psi^*_{RE} < 0 \\ RE_x(z,y,t,c) - 4(1 - A(x,z,t,c)) & \text{if } \Psi^*_{RE} \geq 0 \end{cases}$$

Because monitoring is done every nth transaction, Ψ^*_{RE} is equal to the Ψ_{RE} that was obtained at the last transaction event.

Let $ITL_x(y,t,c)$ denote the *instantaneous trust level* (ITL) of y obtained by x as a result of monitoring its current transaction with y for context c at time t. The ITL is determined based on a single transaction between x and y. In practice, the monitoring process can be done offline, online, or combining offline and online mechanisms. The *transaction monitor* (TM) proxy of x (i.e. TM_x proxy) determines the $ITL_x(y,t,c)$ of the transaction. Because a TM proxy is controlled by the associated peer, a transaction can be rated to have different trust levels by different TM proxies. TM proxies observe the transaction or the transaction records to determine whether any abuses have taken place.

Monitoring the transactions in a real-time *Intrusion Detection Systems* (IDSs) automate the cumbersome task of going through the rather jungle-like audit data trails. Monitoring each transaction is an onerous task. Therefore, the monitoring process (i.e., obtaining $ITL_x(y,c,t)$ is done every nth transaction. After the transaction is over and if x carried out the monitoring process (i.e., $ITL_x(y,c,t)$ is obtained), then $\Psi_{RE_x}(z,y,t,c)$ can be calculated as shown in Equation 2, where $\Psi_{RE_x}(z,y,t,c)$ is z's recommendation error as observed by x when z gives recommendation regarding y for context c at time t.

$$\Psi_{RE_x}(z,y,t,c) = RE_x(z,y,t,c) - ITL_x(y,t,c) \quad (2)$$

The value of $|\Psi_{RE_x}(z,y,t,c)|$ is an integer value ranging from 0 to 4 because $RE_x(z,y,t,c)$ and $ITL_x(y,t,c)$ are in [1..5]. Notice, that $\Psi_{RE_x}(z,y,t,c)$ is computed if and only if $ITL_x(y,t,c)$ is obtained. Hence, the accuracy of z when giving recommendation regarding y for context c at time t as far as x is concerned (i.e., $A_x(z,y,t,c)$) can be defined by:

$$A_x(z,y,t,c) = -\frac{1}{4} |\Psi_{RE_x}(z,y,t,c)| + 1 \quad (3)$$

Note that, $A_x(z,y,t,c)$ is a real number in the interval $[0,1]$. If $|\Psi_{RE_x}(z,y,t,c)| = 0$, $A_x(z,y,t,c) = 1$ implying that z has maximum accuracy as far as x is concerned. Inversely, if $|\Psi_{RE_x}(z,y,t,c)| = 4$, $A_x(z,y,t,c) = 0$ meaning that z is completely inaccurate to x about y as far as x is concerned. Note that $A_x(z,y,t,c) = 0$ if and only if $|\Psi_{RE_x}(z,y,t,c)| = 4$ meaning that: (a) $ITL_x(y,t,c) = 1$ and $RE_x(z,y,t,c) = 5$ or (b) $ITL_x(y,t,c) = 5$ and $RE_x(z,y,t,c) = 1$. This is because the largest recommendation error results in the lowest accuracy and the smallest recommendation error results in the maximum accuracy

Note also that, $A_x(z,y,t,c)$ is the accuracy of z pertaining to the current particular transaction between x and y. Whereas, $A_x(z,t,c)$ is the overall accuracy of z, regardless of the target peer, as far as x is concerned. Finally: (a) $A_x(z,y,t,c)$ will be used to update z's overall accuracy (i.e. $A_x(z,t,c)$), and (b) $\Psi_{RE_x}(z,y,t,c)$ is used to update z's overall recommendation error (i.e., $\Psi_{RE_x}(z,t,c)$). The update procedures are explained in more detail in Section II-D.

C. Computing Trust and Reputation

In computing trust and reputation, several issues have to be considered. First, the trust may decay with time. For example, if x trusts y at level p based on past experience five years ago, the trust level today is very likely to be lower unless they

have interacted since then. Similar time-based decay also may apply for reputation. Second, entities may form alliances and as a result would tend to trust their allies more than they would trust others. Finally, the trust level that x holds about y is based on x's direct relationship with y as well as the reputation of y, i.e., the trust model should compute the eventual trust based on a combination of direct trust and reputation and should be flexible to weigh the two components differently.

Let the behavior trust for a given context c and time t between two entities x and y be $\Gamma(x, y, t, c)$, direct trust between the entities for the same context and time be $\Theta(x, y, t, c)$, and the reputation of y for the same context and time be $\Omega(y, t, c)$.

Trust is allowed to change with time and the number of experiences. For example, if x trusted y at a given level five years ago, x's trust in y now is likely to be lower unless x and y have continued to interact since then. To model this aspect, we multiply the trust levels in DTT_x by a *decay function* $(\Upsilon(t - \tau_{xy}((t), c)), TF_{xy}(c))$, where t the current time, $\tau_{xy}((t), c)$ is the time of the last transaction between x and y for context c, and $TF_{xy}(c)$ is the transaction frequency between peer x and peer y for context c.

$$\Gamma(x, y, t, c) = \alpha\ \Theta(x, y, t, c) + \beta\ \Omega_x(y, t, c) \quad (4)$$

$$\Theta(x, y, t, c) = \Upsilon(t - \tau_{xy}((t), c), TF_{xy}(c))\ DTT_x(y, t, c) \quad (5)$$

The reputation of y is computed as the average of the product of the trust level in the DTT shifted by the *shift function* S and the *decay function* $(\Upsilon(t-\tau_{zy}(t), c))$, for all recommenders $z \in R$ and $z \neq y$. This is indicated in Equation 6. In practical systems, entities will use the same information to evaluate direct trust and give recommendations, i.e., DTT will be used to give recommendations as well as for obtaining the direct trust level.

$$\Omega_x(y, t, c) = \frac{1}{|R_x|} \sum_{z \in R_x} \Upsilon(t - \tau_{zy}((t), c), TF_{xy}(c))\ S(x, y, z, t, c) \quad (6)$$

D. Trust Evolution

Suppose that based on the trust evaluations peer x decides to go ahead with the transaction, the transaction can be monitored by the TM_x and the TM_x proxies. The TM_x and the TM_y proxies determine $ITL_x(y, t, c)$ and $ITL_y(x, t, c)$, respectively. Because a TM proxy is controlled by the associated peer, TM proxies of x and y might evaluate the same transaction differently. For how a transaction is monitored and examples of conditions that can cause a breach in the transaction between x and y, please refer to Section II-B.

In the remaining of this section, we detail how the updates are carried out by peer x. If ITLs are obtained by the TM proxies, then they are used to evaluate the two sources of information regarding trust between x and y. For example, if ITL_x is obtained, the following two sources of information are updated: (a) direct trust between x and y (i.e., $DTT_x(y, t, c)$) and (b) the accuracy of recommender z in making recommendations for context c at time t (i.e., $A_{RTT_x}(z, t, c)$

The two sources are evaluated differently. Let $ITL_x(y, t, c)$ denote y's ITL for context c at time t as observed by x and $DTT_x(y, t, c)$ be the trust level of the DTT_x entry that corresponds to the level x trusts y for context c at time t based on direct interaction that x had with y. Let δ be a real number between 0 and 1.

$$DTT_x(y, t, c) = \delta\ DTT_x(y, t, c) + (1 - \delta)\ ITL_x(y, t, c) \quad (7)$$

If $\delta > 0.5$, preference is given to the ITLs determined through the analysis of the previous transactions between x and y.

To evaluate the set of recommenders, x needs to compute the consistency as well as the accuracy measures as shown in Section II-B. Suppose $\Psi_{RE_x}(z, y, t, c)$ the recommendation error for recommender z based on the recommendation regarding y that z gave to x for the current transaction and $\Psi_{RE_x}(z, t, c)$ is the accuracy of recommender z maintained based on all previous recommendations made by z. The following formula is used to update z's recommendation error.

$$\Psi_{RE_x}(z, t, c) = \delta\ \Psi_{RE_x}(z, t, c) + (1 - \delta)\ \Psi_{RE_x}(z, y, t, c) \quad (8)$$

Suppose $A_x(z, y, t, c)$ is the accuracy of recommender z based on the recommendation regarding y that z gave to x for the current transaction and $A_{RTT_x}(z, t, c)$ is the accuracy of recommender z maintained at RTT_x based on all previous recommendations provided by the average accuracy measure. The following formula is used to update the average accuracy measure.

$$A_{RTT_x}(z, t, c) = \delta A_{RTT_x}(z, t, c) + (1 - \delta)A_x(z, y, t, c) \quad (9)$$

The above equations use a weighted averaging scheme to determine the update for the parameters. In Section III-B, we show how fuzzy logic can be used to estimate the weights (i.e., δ's) and to update the parameters.

The consistency for the recommenders is updated differently. Suppose $C_x(z, y, t, c)$ is the consistency of recommender z based on the recommendation z gave for the current transaction to x regarding y for context c at time t.

Let $C_{RTT_x}(z, t, c)$ denote the consistency of recommender z maintained at RTT_x based on all previous recommendations provided by z. The following simple formula updates $C_{RTT_x}(z, t, c)$.

$$C_{RTT_x}(z, t, c) = \min(C_{RTT_x}(z, t, c), C_x(z, y, t, c)) \quad (10)$$

III. Fuzzy Trust

A. Trust Representation

Behavior trust is quantified by the dynamic parameter *trust level* (TL) that ranges over a set of linguistic label values from *very untrustworthy* to *very trustworthy* as illustrated in Table I. For each trust level, we associated a *Triangular Fuzzy Number*

TABLE I
DESCRIPTION OF THE FUZZY TRUST LEVELS.

Trust Level	Description	TFN
VL	very untrustworthy	[-1.25,0,1.25]
L	untrustworthy	[0,,1.25,2.5]
M	medium trustworthy	[1.25,2.5,3.75]
H	trustworthy	[2.5,3.75,5]
VH	very trustworthy	[3.75,5,6.25]
U	unknown	[0,0,0]

(TFN) that enables us to specify a range for a given trust level instead of giving it a particular discrete value. A TFN is defined as a triplet (a_1, a_2, a_3) where $a_1 <= a_2 <= a_3$. The membership function (μ) of a TFN is defined as follows:

$$\mu(x) = \begin{cases} 0 & \text{if } x = a_1 \text{ or } x = a_3 \\ 1 & \text{if } x = a_2 \\ \frac{x-a_1}{a_2-a_1} & \text{if } a_1 < x < a_2 \\ \frac{x-a_3}{a_2-a_3} & \text{if } a_2 < x < a_3 \end{cases}$$

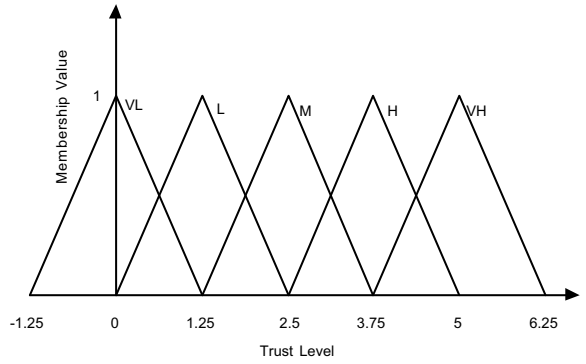

Fig. 1. Triangular Fuzzy Numbers Used in the model.

Suppose that there are two linguistic labels (i.e., a and b) which are represented by using TFNs; then some operations are defined as follows [7]:

1) Addition: $(a_1, a_2, a_3) + (b_1, b_2, b_3) = (a_1 + b_1, a_2 + b_2, a_3 + b_3)$
2) Subtraction: $(a_1, a_2, a_3) - (b_1, b_2, b_3) = (a_1 - b_1, a_2 - b_2, a_3 - b_3)$

We use symmetric TFN with equal width of 2.5. As illustrated in Figure 1, the lowest TFN is centered on a value of 0, and the highest TFN is centered on a value of 5. We used TFNs because of the following reasons: First, it provides simplicity of implementation with just two parameters (i.e., the center point and the distance of the extreme points from the center point) [7]. Second, applying elementary fuzzy arithmetic operations (e.g., addition) yields TFNs without any complications [7].

B. Trust Aggregation and Evolution

The aggregation process is done according to equation 11, which is basically the fuzzy behaviour trust equivalent to equation 4.

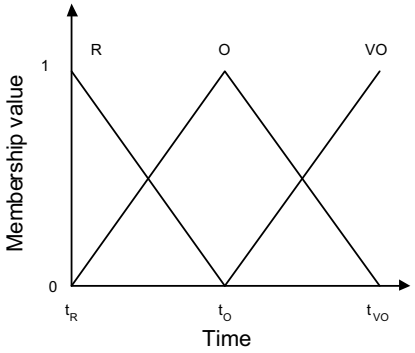

Fig. 2. Recommendation age membership functions.

TABLE II
INFERENCE RULES FOR TRUST AGGREGATION.

Rule #	Description
1	If $(t - \tau_{zy}((t), c))$ is RECENT and $TF_{zy}(c)$ is HIGH, then w_i is VERY HIGH
2	If $(t - \tau_{zy}((t), c))$ is RECENT and $TF_{zy}(c)$ is MEDIUM, w_i is HIGH
3	If $(t - \tau_{zy}((t), c))$ is RECENT and $TF_{zy}(c)$ is LOW, w_i is MEDIUM
4	If $(t - \tau_{zy}((t), c))$ is OLD and $TF_{zy}(c)$ is HIGH, w_i is HIGH
5	If $(t - \tau_{zy}((t), c))$ is OLD and $TF_{zy}(c)$ is MEDIUM, w_i is MEDIUM
6	If $(t - \tau_{zy}((t), c))$ is OLD and $TF_{zy}(c)$ is LOW, w_i is LOW
7	If $(t - \tau_{zy}((t), c))$ is VERY OLD and $TF_{zy}(c)$ is HIGH, w_i is MEDIUM
8	If $(t - \tau_{zy}((t), c))$ is VERY OLD and $TF_{zy}(c)$ is MEDIUM, w_i is LOW
9	If $(t - \tau_{zy}((t), c))$ is VERY OLD and $TF_{zy}(c)$ is LOW, w_i is VERY LOW
10	If a given trust level is UNKNOWN, w_i is zero

$$\Gamma(x, y, t, c) = \frac{1}{\sum_{i=0}^{|R_x|}(w_i)} (w_0 \Theta(x, y, t, c) + \sum_{j=1}^{|R_x|}(w_j \Omega_x(y, t, c))) \quad (11)$$

The objective now is to determine w_0 (i.e. α) and w_j's (i.e., β's). As stated in Equation 6, peer x obtains the reputation of peer y from all z in R_x. The reputation vector from each recommender contains: (a) the recommended TL (i.e., $RE_x(z, y, t, c)$, (b) the recommendation time stamp (i.e., $\tau_{zy}((t), c)$, and (c) transaction frequency (i.e., $TF_{zy}(c)$). The direct trust vector that x obtains from DTT_x contains the same fields as the reputation vector.

$RE_x(z, y, t, c)$ is adjusted with shift function illustrated in Section II-C. Then, the reputation of y is calculated by

aggregating the recommendation as illustrated by Equation 6. Weights are assigned to the recommendations based on their $\tau_{zy}((t), c)$ and $TF_{zy}(c)$ using fuzzy inference rules as in Table II. The inputs to the inference rules are: (a) For each recommendation $t-\tau_{zy}((t), c)$ is calculated and fuzzified with membership functions shown in Figure 2, where three fuzzy sets are used (i.e., RECENT (R), OLD (O), and VERY OLD (VO)), and (b) Each $TF_{zy}(c)$ is also fuzzified using three fuzzy sets (i.e., LOW (L), MEDIUM (M), and HIGH (H)) with membership functions shown in Figure 3. The weights w_0 (i.e., α) is obtained in a similar manner as the recommendation weights.

For the trust evolution, we do the following. As illustrated in Equation 7, we need to assign a weight (i.e., δ) to update DTT_x. These weights are assigned based on their $\tau_{xy}((t), c)$ and $TF_{xy}(c)$ using fuzzy inference rules similar to II. The weights (i.e., δ's) for Equations 8 and 9 are determine in a similar fashion.

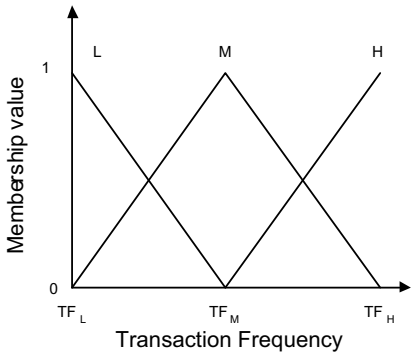

Fig. 3. Transaction frequency membership functions.

IV. Related Work

We discuss several papers that examine issues that are peripherally related. We start by discussing papers that model direct trust and reputation without using fuzzy logic. Then, we examine some papers that use fuzzy logic to tackle trust and reputation modeling.

A trust management in a P2P information system is proposed in [1], where the focus is on implementing a generic scalable infrastructure to deploy any trust model. A simple trust model was proposed, where peers file complaints based on bad experiences they had while interacting with other peers. One limitation of this model is that it is based on a binary trust scale (i.e., a peer is either trustworthy or not).

Assuming that less than 50% of a population of peers are malicious, a simple reputation polling mechanism is presented in [9]. In this recommendation scheme, a peer's trustworthiness is determined through majority voting.

Fuzzy techniques are proposed to manage direct trust and reputation in [2], where a peer uses a polling protocol to query and select target peers Each peer maintains information on its own experience with target peers and shares such experiences when polled by other peers. This approach has no mechanism to filter out dishonest peers from the reputation network. A peer broadcasts its request to all of its neighbors regardless of their honesty.

Paper [10] has developed a system for evaluating peer reputation. This system is targeting P2P e-commerce applications. It performs fuzzy logic inferences to determine the local trust scores and uses accumulated local scores for weight inference in global reputation aggregation. Local scores are generated by performing fuzzy inference on local parameters. Our system leaves the local score computation totally upon the local peer and does not attempt to model its calculation. In [10], global reputation aggregation is done by giving weights to the local scores collected from other peers.

V. Conclusions and uture Work

In this paper, we used fuzzy logic to model trust since the trust is not simply a black and white notion. The qualitative approach of the fuzzy logic is very useful because it is intuitive to start the process with natural language labels that represent intervals rather than exact values. This paper enhances on our previous work in three folds: (a) redefining the honesty concept and its usage by differentiating between consistency and honesty, (b) utilizing the decay function and including two input parameters, namely the time stamp and the transaction frequency, and (c) extensively using fuzzy logic to model trust representation, trust aggregation, and trust evolution. By using fuzzy logic to determine the weights for direct trust as well as reputation, our fuzzy trust model becomes flexible to rely on direct trust or on reputation. We are currently, carrying out simulation experiments to measure the performance of our fuzzy trust model with incorporating consistency and utilizing the decay function.

References

[1] K. Aberer and Z. Despotovic. Managing trust in a peer-2-peer information system. In *10th International Conference on Information and Knowledge Management (CIKM'01)*, pages 310–317, Nov. 2001.

[2] R. Aringhieri, E. Damiani, S. D. Vimercati, S. Paraboschi, and P. Samarati. Fuzzy techniques for trust and reputation management in anonymous peer-to-peer systems. *Journal of the American Society for Information Science and Technology*, 57(4):528–537, 2006.

[3] F. Azzedin, M. Maheswaran, and A. Mitra. Applying a trust brokering system for resource matchmaking in public-resource grids. *Journal of Grid Computing*, To appear., 2006.

[4] R. Dingledine, M. J. Freedman, and D. Molnar. Accountability. In A. Oram, editor, *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*, pages 271–340. O'Reilly and Associates, Sebastopol, CA, 2001.

[5] I. Foster and A. Iamnitchi. On death, taxes, and the convergence of peer-to-peer and Grid computing. In *2nd International Workshop on Peer-to-Peer Systems (IPTPS03)*, Feb. 2003.

[6] N. Griffiths, K.-M. Chao, and M. Younas. Fuzzy trust for peer-to-peer systems. In *P2P Data and Knowledge Sharing Workshop (P2P/DAKS 2006)*, page To appear, July 2006.

[7] M. Marimin, I. Hatono, and H. Tamura. Linguistic labels for expressing fuzzy preference relations in fuzzy group decision making. *IEEE Trans. Systems, Man and Cybernetics*, 28(22), 1998.

[8] A. Oram, editor. *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*. O'Reilly and Associates, Sebastopol, CA, 2001.

[9] S. Sen and N. Sajja. Robustness of reputation-based trust: Boolean case. In *1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02)*, pages 288–293, July 2002.

[10] S. Song, K. Hwang, R. Zhou, and Y. Kwok. Trusted p2p transactions with fuzzy reputation aggregation. *IEEE Internet Computing*, 9(9):24–34, 2005.