Dubious Feedback: Fair or Not?

Jinsong Han, Yunhao Liu
HongKong University of Science and Technology
{jasonhan, liu}@cs.ust.hk

Abstract

Reputation based trust management is increasingly popular in providing a quantitative measurement for peers choosing reliable resources and trusted cooperators in a decentralized Peer-to-Peer (P2P) environment. However, existing approaches do little regarding the validation of a peer's reputation, that is, it is challenging to guarantee the validation and accuracy of computing a reputation value due to malicious denigration or overpraising. In this work, we first investigate the impact of this problem. We then propose TruthRep approach, which encourages peers to provide honest feedback by involving the quality of their evaluations of others into computing reputations. We outline the challenging issues of this design, and present preliminary experimental results.

1. Introduction

Peer-to-Peer (P2P) networks have become popular recently due to their outstanding features such as scalability, reliability and their full-utilization of resources [1, 3, 4, 10, 12-14, 17, 23, 26]. However, those open systems suffer from a set of serious problems: Free-riders consume resources but contribute little or nothing to the system; and malicious nodes deliberately spread inauthentic resources or viruses. Even worse, in existing P2P systems "bad" peers are seldom punished for their malign behaviors. Users in P2P systems thereby request reliable services and trustworthy partners when they share resources and cooperate with each other. Recent research [6-9, 19, 24, 25] suggests that reputation based trust management is an appropriate solution to build trust to online transaction participants and their offerings. A reputation assigned to a user is a global view that all other entities refer to before awarding their trust to this specific user. Such a mechanism is able to help users to choose reliable resources as well as improving the cooperation among them in P2P systems.

Unfortunately, some built-in flaws of reputation based trust management impede its implementation. One of the most prominent threats to the disposition of reputation based trust systems is a false report on ones reputation. For example, one or a clique of malicious nodes deliberately denigrates trustworthy nodes and the resources they provide, or purposely overstate the reputation of members within their group. The lack of authentication makes it difficult to differentiate between defaming statements or exaggerated praise and honest feedback in open and distributed P2P systems. If fallacious opinions are involved in the calculation for a specific peer's reputation, the fairness and authority of a trust management system would be badly damaged. Hence, it is a fundamental requirement for all reputation based trust management systems to enhance peers honesty when they provide opinions. Existing approaches have problems as follows.

1. Most systems make an assumption that feedback honestly and impartially represents the real opinions of peers. Feedback collected from untrustworthy peers is barely discernable from that made by trustworthy ones.

2. Most systems lack protection from clique cheating. If a group of malicious peers collaborate to denigrate or overpraise a specific peer, the error from its expected reputation value might be extremely large.

3. Most systems do not combine an incentive mechanism, if used, with the quality of feedback. This drawback allows malicious peers to extricate themselves from punishment for their dishonest feedback.

With the above challenging issues in mind, we propose a reputation based trust management system called TruthRep, to enhance peers to provide truthful feedback. Our approach makes a number of unique contributions:

- We define the feedback validation problem and analyze the key perspectives and its impact. The description of this problem will bring about to an in-depth understanding of the feedback quality issue.
- We sketch a number of challenging issues in this design. We believe that it is necessary for other reputation based management systems to carefully address these problems as well.
- We propose a novel reputation computing model, which allows the feedback quality to impact on each peer's reputation. As a result, peers have to be honest both in interactions with others and in their feedback reporting.

INFOSCALE '06. Proceedings of the First International Conference on Scalable Information Systems, May 29-June 1 2006, Hong Kong

This model also embeds an incentive mechanism to encourage peers to provide truthful feedback to increase their reputation.

The remainder of this paper is organized as follows. Section 2 overviews the related work. Section 3 presents the design of TruthRep protocol. We also discuss relative attacks and challenging issues in Section 3. Section 4 presents our simulation results and evaluations. We conclude this paper in Section 5.

2. Related Work

Reputation based trust management mechanisms have been widely employed in online electronic communities and distributed environments. Most of them largely depend on feedback for the reputation computing. Little attention, however, has been paid to the validation of the feedback. They usually assume the feedback is reliable and unbiased. Therefore, deceptive feedback from malicious nodes can be involved in reputation computation.

For example, e-Bay [2], as a pioneering auction system, assigns each user a unique reputation and builds an intuitive reputation computing model. During transactions, bidders give their evaluations to a vender based on this model. A higher reputation of a certain vender suggests that this user might be reliable and trustworthy in a transaction. Obviously, the effectiveness of this model is dependent on the correctness of the assumption that all users behave honestly.

Reputation thereby is categorized into three types [22]: positive reputation, negative reputation, or a combination of both. The positive reputation mechanisms merely count successful transactions in the reputation computing. Negative reputation mechanisms, on the other hand, distribute negative feedback or complaints among peers. Relying only on a positive reputation or a negative reputation in reputation computing is incomplete for generating a reputation of a peer in a comprehensive way. We adopt a combination of positive and negative reputations in our approach to make the trust mechanism more accurate and reliable.

There are many reputation management systems in the P2P community such as P2PRep [6], XRep [7], EigenTrust [9], and Peertrust [24]. P2PRep and XRep are based on Gnutella protocol, providing a reputation computing service for their nodes. In P2PRep [6], feedback is collected through polls. It employs public key cryptography to authenticate feedback messages in delivery. XRep [7] improves P2PRep by assigning reputation values for both peers and resources. XRep also develops a refined re-check procedure in the voting phase to mitigate the impact of malicious nodes. These approaches suffer from two major shortcomings: they require peers to be authenticated during the delivery of '*TrustVote*' messages, which may compromise the anonymity of peers; the poll mechanism fails to include the feedback from offline peers.

EigenTrust [9] provides a pseudo-global reputation computing model. It assumes that the trust relationship is unconditionally transitive and performs distributed computing based on a virtual global reputation metric. This scheme significantly extends the reputation computing scope in P2P systems. Such a computing model, however, is limited in representing a reputation precisely due to its one dimension measurement of trust value. Also, the assumption that there exists a set of pre-trusted nodes to maintain fair reputation computing in the system is critical. Another problem is that this trust system lacks protection from fake transactions.

Strategic malicious nodes are able to incur a serious problem [24]. For example, a number of peers could cumulate their reputation by behaving in a friendly manner first, and then they behave maliciously to consume their accumulated reputation. Peertrust and Trustguard are two representative approaches aiming to address this issue. Peertrust [24] is devoted to considering the quality of feedback, in terms of 'Feedback Credibility'. It measures the feedback credibility by using the feedback provider's trust value and the similarity of feedback between two different groups. Dishonest feedback can be partially eliminated in Peertrust's framework. Based on the Peertrust platform, Trustguard [19] involves the historical performance and recent sudden-changes in behavior of the node. Although these approaches give careful designs in their computing model, the precision of deciding on feedback credibility is still a challenging issue. We extend the feedback credibility problem to a more general form: the feedback validation problem. Besides mitigating impact from malicious nodes, our approach brings the feedback quality of a peer into its reputation computing.

3. TruthRep Protocol

In this section, we present the design of the TruthRep model.

3.1 Problem Definition and Statement

We first formally define the feedback validation problem in reputation based trust management systems.

In most previous work, a peer's reputation is defined as

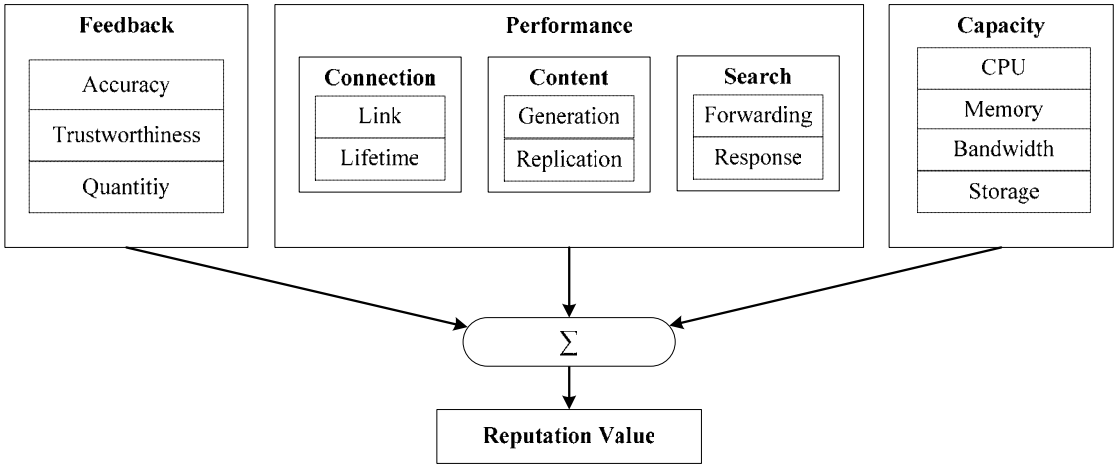

Figure 1. Components of feedback

the global evaluation of its performance and behaviors in historical interactions. We use a quantitative measurement to define the Deviation of feedback as follows.

Definition: Let $\overline{F(x,t)}$ denote the mean of honest feedback for peer x at time t ($-1 \le \overline{F(x,t)} \le 1$). Let $f(u, v, t)$ denote the actual feedback of u for v computed at time t ($-1 \le f(u, v, t) \le 1$). Let the error of a peer u's feedback for peer v at time t as $e(u,v,t) = f(u,v,t) - \overline{F(v,t)}$. Let $\Psi(u, t)$ denote the set of peers interacting with peer u at time t. We define the deviation of e

$$dev(u,t) = \sqrt{\frac{\sum_{i=1}^{\Psi(u,t)} \left(f(u,i,t) - \overline{F(i,t)}\right)^2}{|\Psi(u,t)|}}$$

as the **Feedback Valid Degree** (FVD) of peer u at time t.

FVD is employed to measure the feedback validation of a peer. Let N be the set of total peers in a system, we further define $V(N, t) = \sum_{u \in N} d(u,t)/|N|$ as the validation parameter of a reputation-based trust management system. Higher $V(N, t)$ means more honest feedback peers provide at time t.

In practice, to obtain the precise values of $V(N, t)$ is computationally infeasible in large scale P2P systems. Therefore, we have to use the data colleted by super peers to approximately evaluate the feedback validation. Our goal is to utilize the FVD to leverage peers providing honest feedback and ultimately minimize $V(N, t)$ of the system.

Generally speaking, one peer's reputation can be tampered with in three ways.

- One widely employed method to make the reputation generation untrustworthy is where malicious nodes deliberately provide mendacious statements about one specific resource or node that can impact on the final trust value of this target. The effect becomes more prominent if they collaborate to perform their attacks in the form of a clique.
- Malicious nodes forge opinions of the trustworthy nodes. In this scenario, both vendee and vendor would suffer from reputation degradation due to faked feedback.
- Last, but not least, inactive nodes may only download resources, but not provide feedback on downloaded resources and resource providers. As a result, the final reputation value might be computed based on a small quantity of feedback. The limited feedback sample space causes a large deviation from one correct reputation value.

3.2 Feedback Components

To address the above problems, we design our TruthRep model. Generally, the reputation of a peer is generated based on feedback from all entities that have had transactions with this specific node. Upon our observation, most existing approaches straightforwardly allow peers to generate feedback from the satisfaction degree with respect to the received resource. In fact, a set of important components need to be considered in the feedback generation to represent a comprehensive reflection of

one peer's reputation. We conclude representative components in Fig. 1. In TruthRep, a peer gives feedback for one transaction based on these concrete aspects. Consequently, the feedback is more unbiased. One key point here is how to distribute the weight of these factors in feedback computing. To simplify the prototype design, we only use a simple mean of different factor ratings on which the resource quality and evaluation accuracy have a significant impact. Further discussion about this issue will be conducted in our future work.

3.3 Challenging Issues

TruthRep takes a first look at the involvement of evaluation quality in reputation computing. However, implementing a large reputation based protocol on practical P2P systems is still risky. We discuss some major hurdles in this section and give a deep insight into the difficulties of deploying such protocols in practice.

3.3.1 Evaluation converting

The trust metric is always an essential issue in a reputation based trust management system. In real life, people make an evaluation based on their subjective opinion. An actual reputation, however, is a kind of objective conclusion which should represent the viewpoints of a maximum number of evaluators. Such a reputation usually includes some subtle semantic features, which are too complicated to be represented as quantitative measurements. We have discussed the necessary elements to generate the evaluation. In practice, we need to improve on a semantic based data structure to define the subtle evaluations. A more precise translation from subjective opinions to numerical values leads to more accurate reputation values to reflect overall evaluations.

3.3.2 Malicious clique

Ideally, biased or malicious feedback should be removed from the final reputation computing. Malicious nodes, however, intend to collaborate and protect themselves when providing untrustworthy values. Peers in a malicious clique can perform two collaborating behaviors, one is to co-overpraise the members in their group; another is to deliberately denigrate other peers out of the group.

One promising solution is to introduce a similarity factor to filter malicious cliques. However, it is much difficult to employ similarity based mechanism to deal with this problem because: (1) it is hard to define the similarity factors, in which we need to define subtle metrics to identify clique peers from honest evaluators; (2) a similarity based filter must provide an efficient computing scheme such that the tradeoff between latency and accuracy have to be well balanced.

3.3.3 Free-riding

Another goal of reputation based trust management is to constrain the number of free riders by using their reputation to represent the contribution of peers and rewarding them accordingly. Free-riders rarely provide opinions when consuming resources and services. Consequently, the feedback generated from a limited number of opinions fails to represent a balanced and comprehensive result of a peer's reputation. If comments from malicious nodes become an overwhelming majority in the collected opinions, extremely distorted views about benign peers' reputations might prevail. The core issue of preventing free-riding is that free-riders' behaviors are often legal. Some literature proposed incentive mechanisms to leverage free-riders to contribute more to the system. To our knowledge, there is no proposal considering the free-riding problem in reputation computing. TruthRep takes a first look at this problem. We believe delving more deeply into this area would lead to a number of promising solutions to the free-riding problem.

3.4 TruthRep Architecture

Most popular P2P systems are decentralized and unstructured, in that the peers are interconnected in an ad hoc way [11, 17]. Initiating peers flood queries to request resources. Each query is embedded with a time-to-live (TTL) counter to limit the flooding scope. All peers who forward their queries temporarily store their route information so that every response sent from a responder can be delivered along the reverse paths of these queries. Initiators choose the desired responder and download the resource directly. This model is more reliable and scalable than the centralized client and server models by fully utilizing the resources of all peers. However, the main drawback of the above architecture is that the usage of a flooding search is more inefficient than centralized approaches in which the global resource index is maintained by the centralized server. Meanwhile, a flooding search incurs a large volume of traffic overhead.

To address these problems, hybrid architecture has been implemented in P2P systems to achieve search efficiency [4, 21]. Instead of all peers, only a number of peers, normally with high processing power and capacity, interconnect with others in a decentralized

way in hybrid P2P systems. Each of them, called a super peer, connects some leaf nodes to form a centralized group. In each group, the super node acts as a centralized server, providing index storage and a query service for its members. Super peers reply to queries from leaf nodes if they can find the matching results in their local index lists. Otherwise, they flood the queries among themselves. Hybrid P2P systems keep high reliability and scalability, improve search efficiency from a pure decentralized model, and reduce the traffic overhead significantly.

TruthRep is built over a Gnutella-like hybrid unstructured P2P file sharing system. Our goals behind TruthRep include: (1) providing a comprehensive computation model to generate reputations; (2) involving the quality of evaluations into this model; (3) implementing an incentive scheme to encourage peers to be honest; (4) filtering the feedback from malicious peers and constraining their future behavior.

Super peers, namely agents in our approach, play important roles to achieve the above goals. Besides providing the file index service for their groups, they also manage the reputation of a group of leaf peers. The basic idea is that we introduce some monitoring mechanisms to allow super peers to perform reputation management functions. They report and update the reputations of the dominated nodes, cooperate to locate malicious peers, and encourage all peers to be honest and contribute. We depict TrustRep's architecture in Fig. 2.

3.5 Reputation computing

In our approach, reputation computing consists of three basic phases. First, peers generate and collect the feedback about the evaluated peers. Second, a comprehensive model is used to compute the reputation in an unbiased, honest, and efficient way. Impact from

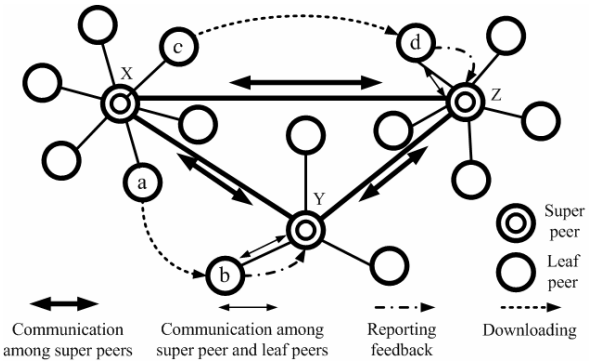

Figure 2. TrustRep architecture

untrustworthy feedback should be minimized. In the last phase, peers make use of their reputation to choose reliable and trusted partners for their transactions. Meanwhile, TruthRep also employs an incentive mechanism to encourage peers to act honestly and contribute more to the system. Furthermore, TrustRep takes into account the peers' performance in contributing and under evaluation when updating their reputations.

3.5.1 Feedback generation and collection

In TruthRep, a transaction comprises an initiator, a number of forwarders (super peers), a list of responders, and several resource providers. A peer and its agent generate feedback for those participants, respectively. The feedback is computed based on fundamental components shown in Fig. 1. In detail, leaf nodes create the feedback for responders and resource providers, and agents provide the evaluation towards forwarders. To justify views shown in the feedback, peers need to keep evidence or proof of transactions to support their evaluation. Peers construct proof of transactions by using a public key cryptography based scheme. Let every node x have a public key and a private key (PK_x, SK_x). The public key PK_x is associated with x's ID (say a pseudonym). We assume that each peer uses its private key to create a signature of the trading content and timestamp for each transaction as a proof.

Suppose peer u and v perform a transaction tr_i at time t. Peer u can give its feedback about v's reputation according to the evaluation history, denoted by $f(u, v, t)$. Peer u reports this evaluation to its agent. The agent A aggregates feedback towards v from both u and other peers in this group at time t, denoted by $g(A, v, t)$, and then generates an intuitively integrated evaluation $F_A(v, t)$. Regarding security concerns, TruthRep achieves authentication and integrity through the public key based scheme in the message transmission. In our model, $F_A(v, t)$ can be obtained by any of the existing trust evaluation mechanisms such as [9, 19]. Agent A also keeps $F_A(v, t)$ for each u in its group. To simplify the implementation, we use an intuitive computing model of $F_A(v, t)$ in Equation (1). Component $C_A(v)$ expresses the evaluation to the capacity of peer v from A's point of view.

$$F_A(v,t) = \lambda \cdot \frac{\sum_{u \in g(A,v,t)} f(u,v,t) \cdot R(u,t)}{|g(A,v,t)|} + \mu \cdot C_A(v) \quad (1)$$

In theory, the maximum fairness and objectivity of reputation computing could be achieved if we allow each agent to compute the reputation for all leaf peers,

either in its group or in others, and aggregate all results into Eqs (1). In practice, it is impossible and impractical to have enough storage and computing capacity to support such an ideal model. We need to develop a distributed computation structure to assign a set of agents, called *consultants*, to each peer for its reputation query requirement. If a peer x needs the reputation of peer y, it first queries its consultants, and its consultants return their aggregated feedback about y to x. Peer x can easily compute an approximate global-reputation of y. In TruthRep, we use multiple Distributed Hash Tables (DHTs), like in CAN [18] or Chord [20], to organize peers into structured overlay networks, and assign each trust data item to a specific peer using the hash value of this data item.

For one DHT, a peer's hash value can be computed by hashing a unique ID of the peer, such as its IP address, TCP port, and public key. This value is deterministically mapped to a point in the hash space, and covered by an agent peer. Multiple hash values of a peer accordingly associate it with multiple agents. Those agents are appointed as the consultants of that peer. In this way, both leaf nodes and agents have a set of consultants to provide a reputation query service.

3.5.2 Reputation model

TruthRep uses a comprehensive model consisting of two components to compute the ultimate reputation for each node. Let $R(v, t)$ be the reputation value of a given node at time t. Let $F(v, t)$ be the basic evaluation derived from the feedback about this node, which is the first component of $R(v, t)$. The second component $E(v, t)$ considers the quality of the feedback this peer makes about other peers or resources at time t.

$$R(v, t)=\alpha \cdot F(v, t)+\beta \cdot E(v, t) \quad (2)$$

Constants α and β are used to balance the contribution of three components. Let $Con(v)$ be the set of v's consultants. Then the first part of $R(v, t)$ can be aggregated from the feedback of v's consultants:

$$F(v,t)=\frac{\sum_{X \in Con(v)} F_X(v,t) \cdot R(X,t)}{|Con(v)|} \quad (3)$$

Assume for a given node v, each of its consultants keeps the evaluations made by v about all peers that have traded with v at time t, and thereby holds a FVD $dev(v, t)$ for v at time t. Let Y be a consultant of peer v, we include FVD factor into the computing for feedback quality and compute the $E(v, t)$ in the following way:

$$D_Y(v,t)=K_p dev(v,t)+\frac{K_i}{t}\int_0^t dev(v,\omega)d\omega+K_d dev'(v,t) \quad (4)$$

$$E_Y(v,t)=\frac{\pi}{t}\int_0^t Q_Y(v,t)dt-\rho D_Y(v,t) \quad (5)$$

$E_Y(v, t)$ is composed of two components: the feedback volume $Q_Y(v, t)$ and feedback deviation $D_Y(v, t)$. $Q_Y(v, t)$ represents the quantity of peer v's opinions at time t, while $D_Y(v, t)$ focuses on the computing effect on feedback deviation. $Q_Y(v, t)$ involves the contribution of feedback quantity into v's reputation computing. By taking the feedback deviation into account, TruthRep spurs reasonable evaluation and honest feedback. In the component computing procedure, we use the Proportional-Integral-Derivative (PID) process controller [16] to generate the major deviation of a peer's feedback. The three constants K_p, K_i, and K_d control and balance the contribution from current deviation, historical deviation, and sudden deviation change in recent past, respectively. This controller provides a comprehensive reflection of a peer's feedback quality and imposes the result on its reputation computation. Thus, the second part of $R(v, t)$ can be computed from the feedback of v's consultants:

$$E(v,t)=\frac{\sum_{X \in Con(v)} E_X(v,t) \cdot R(X,t)}{|Con(v)|} \quad (6)$$

3.5.3 Usage of reputation

Each pair of trading parties uses the reputation to direct their transactions. A resource requester can refer to responders' reputations when choosing a desired resource provider. Responders and resource providers also determine whether they would like to reply to a query or deliver the resource to the requester. For example, a vendee retrieves a vendor's reputation values from the vendor's consultants through DHTs. Then the vendee aggregates the final reputation value of the vendor by using Eqs (3) and (6), and make decisions based on the result. After completing the transaction, the vendee and vendor generate their evaluations about the other side and report opinions to their consultants.

Consultants periodically perform reputation computing for their dominated peers. Due to the limitation of storage space, consultants only keep recent computing results in their local storage.

Unfortunately, the received reputation is usually not the updated one due to the system latency and transmission delay. How to provide a real time reputation report is a challenging issue of future

research. In our prototyped model, we simply assume system latency and transmission delay are negligible.

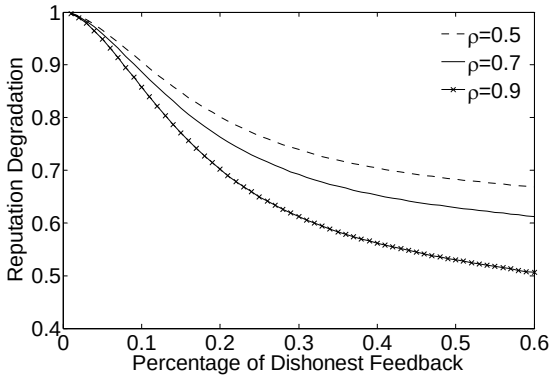

Figure 3. Reputation degradation v.s. percentage of malicious feedback

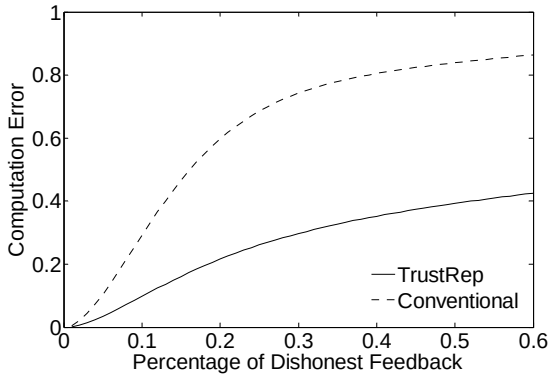

Figure 4. Reputation computing error v.s. percentage of dishonest feedback

Another important issue here is how to combine the incentive mechanism in TruthRep. An intuitive incentive mechanism has been embedded into the TruthRep model shown in Eqs (4) and (5). Peers will be rewarded for their high quality feedback and punished for that of low-quality.

4. Simulation and Evaluation

We conduct some preliminary simulation based on a real P2P trace [21]. The network size ranges from 10^3 to 10^4 nodes. To simulate the underlay below the P2P overlay, we used BRITE [15] to generate a 30,000-nodes internet like topology. In our simulation, resource and query popularity follow a Zipf-like [5] distribution (aka Power Law), where the probability of a request for the *i*th most popular content is proportional to $1/i^{\alpha}$ (α typically is a value less than unity). We repeatedly performed 1000 transactions. In each transaction, a random-selected initiating peer sends out a transaction request and a certain number of peers respond. The initiator and responder of each transaction are arranged in different groups. The two peers then perform the transaction based on the reputation of opposing side to decide whether continue or terminate the transaction. After completing the transaction, two participants report the feedback to their consultants.

Among these peers, some are malicious, some are free-riders, and others are trustworthy. A free-rider peer may also be malicious in some transactions. We set the percentage of malicious nodes to be 0%~10%, and the free-riding nodes to be 0%~50%. The malicious peers may spread dishonest feedback and free-rider give no feedback for their transactions. The percentage of dishonest feedback can reach up to 60% in our experiment. We investigate the average reputation degradation of malicious peers who provide dishonest feedback in Fig. 3. The results show that a peer is punished by degrading its reputation if it provides dishonest feedback. When we enlarge the system parameter ρ, which means we increase the impact of the feedback quality factor in our reputation computing, this effect will be more obvious.

We also examine the computation error in peer reputations when changing the percentage of dishonest feedback. In this experiment, we check out the average reputation computation error according to TrusthRep and the conventional trust computing model. The conventional approach we simulated only utilizes the basic feedback aggregation without considering the feedback quality factor. The primary results in Fig. 4 indicate that our TruthRep outperforms previous work. The prototyped TruthRep reduces the computing error by half when the percentage of dishonest feedback reaches 60%. As a result, peers in TruthRep can obtain more accurate knowledge to direct their transactions.

5. Conclusion

We have presented TruthRep, a reputation based trust management system which aims to address the feedback quality problem in decentralized P2P systems. We propose a coherent reputation computing model which includes the feedback quality of peers. Our study of TruthRep is in its early stages and there are several issues that need to be discussed and studied in the future, such as

improving the anonymity of peers, and developing an integrated trust-incentive management protocol for current P2P applications like BitTorrent, eDonkey / Overnet, and eMule.

6. Acknowledgement

This work is supported in part by Hong Kong RGC DAG05/06.EG44, NSFC No. 60573053, and Microsoft Research Asia.

7. References

[1] BitTorrent, http://www.bittorrent.com/

[2] eBay website, http://www.ebay.com

[3] Gnutella, http://gnutella.wego.com/

[4] KaZaA, http://www.kazaa.com

[5] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web Caching and Zipf-like Distributions: Evidence and Implications", In Proceedings of IEEE INFOCOM, 1999

[6] F. Cornelli, E. Damiani, S. D. C. d. Vimercati, S. Paraboschi, and P. Samarati, "Choosing Reputable Servents in a P2P Network", In Proceedings of the 11th international conference on World Wide Web(WWW), 2002

[7] E. Damiani, S. Vimercati, S. Paraboschi, P. Samarati, and F. Violante, "A Reputation-Based Approach for Choosing Reliable Resources in Peer-to-Peer Networks", In Proceedings of the 9th ACM Conference on Computer & Communication Security, 2002

[8] M. Gupta, P. Judge, and M. Ammar, "A Reputation System for Peer-to-Peer Networks", In Proceedings of the 13th ACM International Workshop on Network and Operating Systems Support for Digital Audio and Video, 2003

[9] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The EigenTrust Algorithm for Reputation Management in P2P Networks", In Proceedings of The 12th International Conference on World Wide Web(WWW), 2003

[10] X. Liao, H. Jin, Y. Liu, L. M. Ni, and D. Deng, "AnySee: Peer-to-Peer Live Streaming", In Proceedings of IEEE INFOCOM, 2006

[11] Y. Liu, A.-H. Esfahanian, L. Xiao, and L. M. Ni, "Approaching Optimal Peer-to-Peer Overlays", In Proceedings of the 13th Annual Meeting of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), 2005

[12] Y. Liu, X. Liu, L. Xiao, L. M. Ni, and X. Zhang, "Location-Aware Topology Matching in P2P Systems", In Proceedings of IEEE INFOCOM, 2004

[13] Y. Liu, L. Xiao, and L. M. Ni, "Building a Scalable Bipartite P2P Overlay Network", In Proceedings of 18th International Parallel and Distributed Processing Symposium (IPDPS), 2004

[14] Y. Liu, Z. Zhuang, L. Xiao, and L. M. Ni, "A Distributed Approach to Solving Overlay Mismatch Problem", In Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS), 2004

[15] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An Approach to Universal Topology Generation", In Proceedings of the International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunications Systems (MASCOTS), 2001

[16] H. Ozbay, Introduction to feedback control theory, CRC Press, 2000

[17] D. Qiu and R. Srikant, "Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks", In Proceedings of ACM SIGCOMM, 2004

[18] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-addressable Network", In Proceedings of ACM SIGCOMM, 2001

[19] M. Srivatsa, L. Xiong, and L. Liu, "TrustGuard: Countering Vulnerabilities in Reputation Management for Decentralized Overlay Networks", In Proceedings of the 14th World Wide Web Conference (WWW), 2005

[20] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications", In Proceedings of ACM SIGCOMM, 2001

[21] D. Stutzbach and R. Rejaie, "Characterizing the Two-Tier Gnutella Topology", In Proceedings of ACM SIGMETRICS, 2005

[22] G. Suryanarayana and R. N. Taylor, "A Survey of Trust Management and Resource Discovery Technologies in Peer-to-Peer Applications", Technical Report, 2004

[23] C. Wang, L. Xiao, Y. Liu, and P. Zheng, "Distributed Caching and Adaptive Search in Multilayer P2P Networks", In Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS), 2004

[24] L. Xiong and L. Liu, "PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities", IEEE Transactions on Knowledge and Data Engineering (TKDE), 2004

[25] B. Yu, M. P. Singh, and K. Sycara, "Developing Trust in Large-Scale Peer-to-Peer Systems", In Proceedings of the first IEEE Symposium on Multi-Agent Security and Survivability, 2004

[26] B. Zheng, J. Xu, W.-C. Lee, and D. L. Lee, "Grid-Partition Index: A Hybrid Method for Nearest-Neighbor Queries in Wireless Location-Based Services", Very Large Data Base Journal (VLDBJ), 2004