Towards Reliable Reputations for Dynamic Networked Systems

Gayatri Swamynathan[†], Ben Y. Zhao[†], Kevin C. Almeroth[†], S. Rao Jammalamadaka[§]
[†]Department of Computer Science, UC Santa Barbara, CA 93106
[§]Department of Statistics and Applied Probability, UC Santa Barbara, CA 93106
{*gayatri, ravenben, almeroth*}@*cs.ucsb.edu, rao@pstat.ucsb.edu*

Abstract

A new generation of distributed systems and applications rely on the cooperation of diverse user populations motivated by self-interest. While they can utilize "reputation systems" to reduce selfish behaviors that disrupt or manipulate the network for personal gain, current reputations face a key challenge in large dynamic networks: vulnerability to peer collusion. In this paper, we propose to dramatically improve the accuracy of reputation systems with the use of a statistical metric that measures the "reliability" of a peer's reputation taking into account collusion-like behavior. Trace-driven simulations on P2P network traffic show that our reliability metric drastically improves system performance. We also apply our metric to 18,000 randomly selected eBay user reputation profiles, and surprisingly discover numerous users with collusion-like behaviors worthy of additional investigation.

1 Introduction

Recent work has proposed a number of highly scalable peer-to-peer routing protocols [2, 20, 22, 30] that support Internet-scale network applications such as distributed storage systems [19] and content distribution networks [12]. However, reliable protocol operation relies on peer cooperation, a difficult challenge given that these heterogeneous peers are distributed over many distinct networks and administrative domains. The use of cheap and anonymous online identities frees users from the consequences of their actions. Studies have shown that without accountability, peers in these open networks can act selfishly, often disrupting or manipulating the network for personal gain [15, 21].

To limit the impact of selfish users who exploit their online anonymity for personal gain, researchers have proposed introducing limited accountability through the use of reputation systems [1, 14]. By producing a statistical estimate of trustworthiness based on feedback from past transactions, reputation systems can guide peer interactions to reduce risk. They have been deployed on e-commerce sites such as eBay and knowledge forums such as SlashDot and Yahoo Answers. On these centralized platforms, reputation mechanisms have significantly improved system reliability.

When applied to large-scale distributed systems, however, reputation systems and their associated protocols face a significant challenge: user collusion. Reputation systems generally assume that each online identity represents a single user. However, recent work has shown that given the relative low cost of online identities, users often generate multiple "Sybil" identities to gain benefits beyond the fair allocation for a single identity [27]. The Sybil attack, as this is known, also allows these multiple identities to "collaborate" or collude for the good of the user, thereby decreasing the effectiveness of online reputation systems. For example, users can collude to artificially boost the reputation values of one or more peers [15], or falsely accuse well-behaved users of misbehavior. Either way, multiple Sybil identities allow the user to obtain significant advantages over other users with single identities.

Any practical application of reputations at the network level must first address the key challenges of user collusion. In this paper, we seek to address this challenge by augmenting traditional reputation systems with a "reliability metric." Our approach helps users make more accurate decisions based on trust by quantifying the risk that a given reputation value has been affected by collusion or collusion-like behavior. This serves as an estimate of the reputation value's accuracy. As a basis for our metric, we leverage a pair of well-studied mechanisms used in economic studies, the Lorenz curve [16] and the Gini coefficient [5]. Applied to reputations, they characterize how far a peer's per-partner distribution of transactions deviates from the ideal. Using our metric, a user can easily distinguish between reputations generated by truthful transactions and those that might be strongly influenced by user collusion.

The work in this paper makes three key contributions. First, we propose and describe in Section 3 a reliability metric for reputation systems that protects users from user collusion attacks. Second, we evaluate in Section 4 our proposed mechanism via detailed simulations of peer collusion models based on recent measurements of a deployed P2P system [15], and show how our enhancements greatly improve the accuracy of traditional reputation systems under peer collusion. Finally in Section 5, we apply our reputation metric to users of a highly successful online marketplace, eBay. We test the reliability of 18,000 randomly selected eBay reputation profiles. Examination of the results shows a surprisingly large number of users who exhibit "collusion-like" behavior. While it is impossible to determine a user's true intention, users identified using filters based on our reliability metric show highly unusual patterns in their transaction histories. This "circumstantial evidence" is highly indicative of forged transactions, and definitely justifies further investigation.

2 Background and Related Work

Background on Reputation Systems. A traditional reputation system collects and aggregates feedback about the behavior of each network peer into a reputation profile. A service requester R, uses a provider P's reputation profile to determine whether to transact with P. Following the transaction, R gives its feedback of P to the reputation system.

Reputation systems generate trust relationships using one of two approaches. One approach, referred to as *local reputation*, uses only firsthand interactions to evaluate peers. Each peer aggregates its own experiences and does not share its opinion with others. A second and more popular approach, referred to as *global reputation*, computes a peer's reputation by aggregating feedback from all of its past transaction partners. While global reputations are vulnerable to false ratings, they provide significantly more information than local reputations.

Our reputation model shares basic mechanisms common to most reputation systems: we employ global reputations and compute peer trustworthiness as an average of all ratings received by a peer over its lifetime. Each peer in the network has a third-party *trust manager* that maintains its *reputation rating* and *reliability rating* computed from its transaction history. We assume a secure storage and exchange protocol that ensures the integrity of the trust data being stored, updated, and communicated [8, 17]. Other optimizations such as avoiding malicious feedback [23] and making reputations incentive-compatible [13] are orthogonal and complementary to our system.

Related Work on Online Attacks. The Sybil attack occurs in the absence of a centrally trusted party, when a user with sufficient resources can establish a potentially unbounded number of distinct online identities [9, 27], and was first identified in the context of peer-to-peer protocols. A colluding attacker will likely launch a Sybil attack to obtain the number of virtual identities necessary for collusion. Other studies have focused on measuring attacks in deployed peer-to-peer systems [11]. A recent study by Lian et al. found strong evidence of collusion behavior between users of a file-sharing network with the express goal of manipulating the incentive system for personal gain [15].

Related Work on Incentives and Reputations. Prior work has shown that reputation systems, if reliable, can effectively motivate trustworthiness and cooperation [1, 4, 6, 7, 14, 17, 25]. However, designing reliable reputation in the presence of collusion and user dynamics is a challenge.

The EigenTrust algorithm applies a global feedback reputation system for P2P networks, and attempts to address the problem of malicious collectives by assuming pre-trusted peers in the network [14]. Zhang et al. improve eigenvector-based reputations by capturing the amount of PageRank inflation obtained by collusions [29]. They observe that colluding nodes cheat the algorithm by stalling the PageRank random walk in a small web graph, and thus are sensitive to the reset probability of the random walk.

Feldman et al. suggest a "stranger adaptive" strategy to counter selfish behavior under network dynamics [10]. Using histories of recent transactions with strangers, a peer estimates the probability of being cheated by the next stranger, and uses this to decide whether to trust the next stranger. Finally, two recent projects apply reputation systems to practical networks. Yu et al. examined the feasibility of using reputations to establish trust between Internet autonomous systems [28]. Credence associates reputations with individual files in a file-sharing network [24].

3 A Reliability Metric for Reputations

Despite their effectiveness in traditional controlled environments, current reputation systems can be highly inaccurate in large dynamic networks such as online communities and P2P networks. The biggest contributing factor to inaccurate reputation values is the increasing presence of peer collusion behavior. Most online communities lack strong authentication, allowing users to obtain multiple "independent" online identities. Prior work has shown that a user can use these identities to collude and artificially inflate his own reputation to monopolize service, lure users into scams, or otherwise gain performance benefits from the system [15].

We address peer collusion by proposing a "reliability" metric that estimates the *accuracy* of a network reputation. Our metric stems from the observation that reputation values are most accurate when computed from numerous past transactions distributed across many distinct partners.

3.1 Peer Collusion Behavior

Before defining our collusion-resistant metric, we need to first clearly define our collusion attack model. We begin this section by quantifying the potential impact of collusion behavior on system-wide performance. We then describe

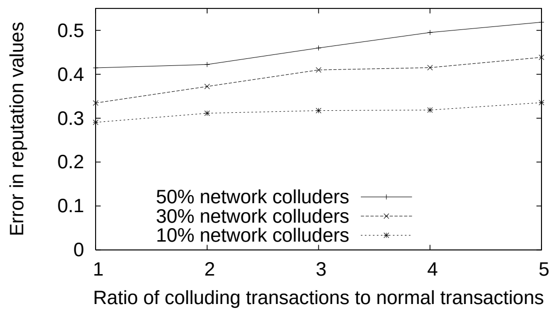

Figure 1. Impact of user collusion on perceived reputations.

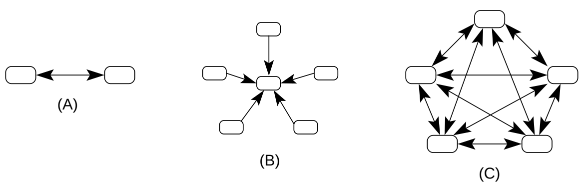

Figure 2. Three different collusion models. (A): pairwise collusion; (B): Sybil-based collusion; (C): group-based mesh collusion.

our assumptions and models for colluding attackers, with models drawn from previous measurement studies.

Impact of User Collusion. To better understand the threat that collusion attacks pose to reputation systems, we perform an experiment using an event-driven simulator where random subsets of a network of 10,000 peers collude to improve their reputation values. We define reputations as values between 0 and 1, where 0 indicates no trust, and 1 indicates absolute trust. For each peer, we define an "intrinsic trust value" that guides the peer in its transactions. For example, a peer with an intrinsic trust value of 0.8 has a random 80% chance of behaving honestly on any given transaction. We set malicious peers with trust values less than 0.3. We then allow random peer pairs to perform transactions in the system, with the subsequent feedback recorded to compute the participants' reputations. We assume a uniform distribution of transactions with an average of 15 normal transactions initiated per peer. In addition to these normal transactions, we allow a subset of 2-5 peers to perform collusion by performing transactions within the group which is always followed by mutual positive feedback. Figure 1 plots the collusion-induced error for affected peers as computed by the difference in reputation values with and without colluding transactions. Clearly, even a relatively low rate of collusion can have a dramatic impact on a peer's perceived reputation values.

Collusion Model. Our collusion model begins with two assumptions. First, we assume that peers cannot modify the application, and must provide verifiable proof of a transaction along with its transaction feedback. This prevents colluders from spoofing an unlimited number of transactions, and can be achieved using reasonable secure signature mechanisms. Second, we assume that while colluders cannot forge transactions, they can perform collusion transactions with resource costs lower than legitimate transactions. For example, data transfers between two application instances on the same machine generally incur much lower processing and I/O overhead compared to typical transactions between distant peers. To model the lower cost of collusion transactions, we use a *collusion cost factor* to represent the ratio of resource costs between a legitimate transaction and a colluding transaction. We use this factor to estimate the number of illegitimate transactions that can be reasonably performed by colluders in our experiments.

To accurately evaluate our metric, we require a test framework with realistic models of user collusion. For this purpose, we leverage the results of a recent measurement study on the Maze peer-to-peer file-sharing network that showed user behavior strongly indicative of multi-user collusion. Maze is a popular file-sharing system in Asia, and uses a centralized architecture that logs all transactions, crediting users for each successful file upload while consuming credits for downloads based on file size [26].

This study [15] examined a complete log of the Maze system over a period of one month, including 32 million file transfers totaling more than 437 terabytes between 161,000 users. It observed several types of highly probable collusion-like behavior, including how multiple peers performed repetitive or faulty transactions to artificially inflate the download credits of certain peers. The results support the prevalence of three popular collusion models. We use these models to drive the test framework used in Section 4. We illustrate these models in Figure 2:

- *Pairwise Collusion.* The simplest model where two peers collude to mutually boost reputation values, e.g. repeatedly download the same content from each other. This can be performed by two distinct users, or by two Sybil identities.
- *Sybil-Based Collusion.* A user boosts its reputation with help from a large number of "slave peers" obtained via a Sybil attack [9]. Slaves exist only to transact with the "master peer" and improve its reputation.
- *Group-Based Mesh Collusion.* Finally, multiple peers can form cliques where all members collaborate to mutually boost reputation values. Peers maximize their benefit by performing pairwise collusion with all other peers in the clique. While the aggregate benefit increases with clique size, clique sizes are limited by non-trivial maintenance and coordination costs.

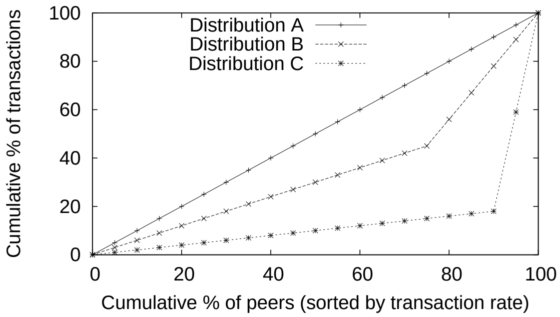

Figure 3. Cumulative Transaction % representation of reputation reliability.

3.2 A Statistical Reliability Metric

To quantify the likelihood that a reputation value has been influenced by possible collusion, we propose a peer reliability metric based on the distribution of transactions among a peer's partner set. A reputation is "less reliable" if a significant fraction of transactions are performed with a small number of peers, and "more reliable" when all transactions are distributed evenly across many distinct partners. Intuitively, we can compute such a reliability by representing a peer *P*'s reputation as a Cumulative Function (CF) of its transaction history. That is, if we plot on the X-axis the cumulative percent of *P*'s distinct partners (sorted by number of transactions undertaken with *P*) and on the Y-axis the cumulative percent of *P*'s transactions, then the most reliable distribution is represented by the 45 degree line.

Figure 3 plots transaction distributions of 3 peers that each conduct 100 transactions with 20 peers. A peer maximizes its reputation reliability by spreading its transactions evenly across all 20 peers in the system (shown by distribution *A*). A colluder who performs 82% of its total transactions with two colluding partners obtains a much lower reliability value for the same total number of transactions (distribution *C*). Finally, an average peer might obtain a partner distribution better than the colluder (distribution *B*).

We investigated the effectiveness of several different measures as potential reliability metrics. Our search led us to the area of economic statistics, where statistical models are used to compute and compare the proportionality of such distributions. The Lorenz curve [16], in particular, is a graphical representation of the cumulative distribution function of a probability distribution. Developed by Max Lorenz in 1905, it is used in economics and ecology to describe inequality in income or size (for example, bottom *X*% of society has *Y*% of the total income). As shown in Figure 4, the Lorenz curve of a given dataset is compared with the *perfect equality line*. In our case, this represents a perfect distribution of transactions among a peer's entire transaction partner set. The further the Lorenz curve lies below the line of equality, the more skewed is the distribution of transactions.

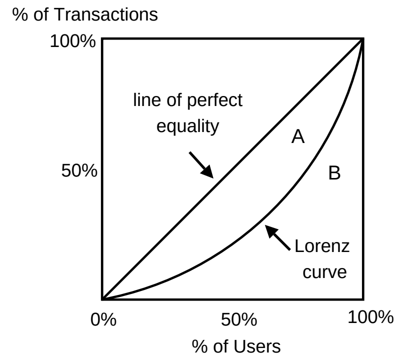

Figure 4. A Lorenz curve graphically represents the proportionality of a distribution.

Formally, the Lorenz curve can be expressed as

$$Z(y) = \frac{\int_0^y x dF(x)}{\mu} \tag{1}$$

where *F(y)* is the cumulative distribution function of ordered individuals and μ is the average size. The total amount of inequality is summarized by the Gini coefficient [5] (*G*). The Gini coefficient of a given data set is the ratio between the area enclosed by the line of equality and its Lorenz curve, and the total triangular area under the line of equality. That is (from Figure 4),

$$G = \left(\frac{A}{A+B}\right) \tag{2}$$

The Gini coefficient ranges between 0 to 1. 0 corresponds to perfect equality, *i.e.* all partners have had the same number of transactions with the given peer. 1 corresponds to maximum inequality, *i.e.* all transactions were undertaken with one single partner. Since higher values are favored by our metric, we compute reliability (or reputation quality) from the Gini coefficient as the following.

$$Q = (1 - G) \tag{3}$$

Here, *Q* denotes a peer reputation's reliability score.

We performed detailed experimental evaluation of this metric in Section 4. Then in Section 5, we use it as an anomaly detection filter to detect misbehaving eBay users from their online reputation profiles.

We note that colluders seeking to boost their aggregate reputation value can easily achieve a high reputation reliability (*Q*) at the same time, by distributing its transactions evenly between its colluding partners. This tactic fails, however, when a colluder actually seeks to make use of its reputation by cheating (and interacting) with a normal user. The more a user colludes with his friends to inflate his reputation, the more significant his drop in reliability after interacting with a non-colluder. In Figure 5, we show how

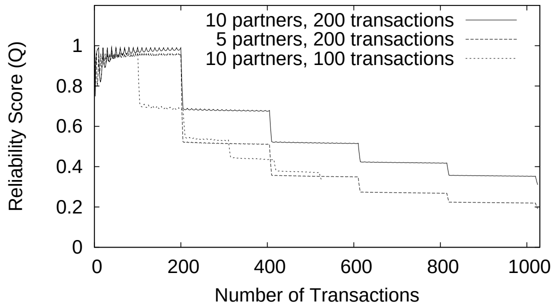

Figure 5. The reliability of an inflated reputation brought down by collusion.

the reliability values of three colluders change as they periodically interact with honest peers. Each colluder starts by building their reputation through collusion, then goes through periodic phases of interacting with normal users followed by more collusion. We compute each colluder's reliability score *Q* after each transaction. During collusion, the colluder cycles through its partner set in a round-robin fashion to evenly distribute its transactions among them.

As the plot shows, transacting uniformly with its partner set produces perfect reliability scores for each user. However, the scores fall dramatically when they interact with non-colluders. Reducing the number of colluding partners or transactions per partner does not result in any improvement in reliability scores of the colluder. Once a reputation's reliability drops, it is hard to re-build afterward. Therefore, a user that colludes frequently with a single partner is permanently damaging his chances for obtaining a high reliability score. Colluders must choose between colluding for higher reputations or spreading out its transactions for a higher reliability score.

4 Performance Evaluation

In this section, we perform detailed evaluation of our reliability metric and demonstrate its role in improving effectiveness of traditional reputation systems. We begin by discussing our simulation setup, including the peer community, reputation schemes employed, and metrics used to evaluate the reputation mechanisms.

4.1 Simulation Setup

Our experiments are performed on an event-driven network simulator of $5,000$ peers. We simulate a large number of peer transactions, where each peer utilizes our reputation framework to choose partners with which to transact. A *transaction* is a two step process: the service requester R, chooses, then performs a transaction with a service provider P. R then assigns P a binary feedback rating of 0 (negative) or 1 (positive). Our "transactions" are general and effectively represent any type of peer-to-peer requests, including financial transactions, information exchange, file read/write or message forwarding operations.

Before each simulation run, we assign each peer a random *intrinsic trust value* between 0 and 1 that represents the rate at which a peer behaves honestly. For instance, an intrinsic trust value of 0.45 means the peer will provide services or ratings honestly with a probability of 0.45. Since colluders are likely malicious peers with low reliability, we set intrinsic trust values for colluders to random values less than 0.30.

Each experiment run includes two distinct phases: a bootstrap phase and an experiment phase. The bootstrap phase initializes peer reputations for all peers. In this phase, each peer performs transactions with random partners, and rates each provider according to its own intrinsic trust value. We fix the number of bootstrap transactions to 10 in our experiments. We assume that colluders can perform more collusion transactions than regular peers, since collusion transactions often consume less resources. We use our *collusion cost factor* parameter (see Section 3.1) to determine the number of collusion transactions undertaken by colluders during the bootstrap phase. For example, a cost factor of 1:1 means colluders can collude at the same transaction rate as normal network peers, while a cost factor of 5:1 means colluders can perform 5 times as many colluding transactions as normal peers.

Once reputation values have been initialized, we begin our experiment phase. In each run, we conduct 150,000 transactions over 5,000 peers for an average of 30 requests per peer. For each transaction, a peer makes a transaction request, and 25 random peers respond. The initiating peer then uses our reputation framework to choose a transaction partner. We use the partner's intrinsic value to determine if the resulting transaction is a success or failure.

Peer Selection Algorithms. To quantify the benefits of our reputation framework, we compare the performance of two reputation systems in our experiments: basic reputations (denoted by *R*) and reputations with reliability metric (*L*). In the basic scheme (*R*), a peer chooses the service provider with the highest reputation value. We compute peer i's reputation value, R_i, as the average of all of its past transaction feedback values. Reputations range between 0 and 1. In the reputations with reliability (*L*) scheme, a peer chooses the provider with the highest weighted combination of reputation and reliability value.

$$L_i = (1 - \alpha) \cdot R_i + \alpha \cdot Q_i \tag{4}$$

Q_i, peer i's reliability score, is computed using Equations 2 and 3. The weight parameter, α, can be tuned by each application to favor higher reputations or more accurate reputations. We set $\alpha = 0.5$ in our experiments.

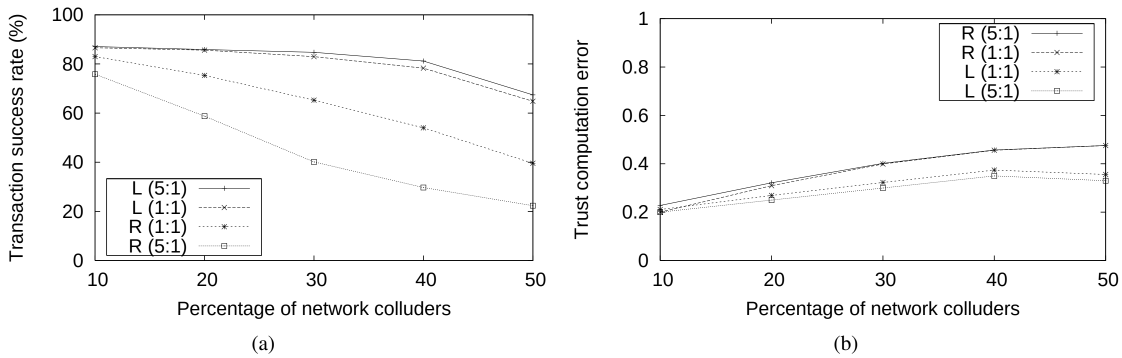

Figure 6. Effectiveness and accuracy against pairwise collusion. R refers to the pure reputations scheme and L refers to the reliability-based reputations scheme. The ratios represent the collusion cost factors.

4.2 Effectiveness, Accuracy and Overhead

We quantify the impact of our reliability mechanisms using three key metrics: transaction success rate, trust computation error, and metric overheads.

Transaction Success Rate. We measure the rate of successful transactions experienced by all peers in the network. A transaction is deemed successful if the provider behaved honestly. The success rate is the ratio of the number of successful transactions over the total number of transactions in the system, and increases when users can avoid untrustworthy partners by making more accurate trust decisions.

Trust Computation Error (TCE). This metric represents how accurately a peer's computed reputation reflects its intrinsic trust value. We use our metric as a relative metric to choose between pairs of partners. We define the TCE in terms of a peer's position in an ordered list of peers sorted by computed reputation. For each reputation system, we compute a sorted list of all network peers based on their reputation values. We then compare this ordered list to the sorted list of all peers based on their intrinsic trust values. A peer's TCE is the difference in its position from one list to the other. For example, if, in a network of 10 peers, the most reliable peer (according to intrinsic trust values) has the third highest computed reputation, its per-peer TCE is $(3-1)/10$. The TCE of a network is the average TCE of all peers, defined as:

$$TCE = \frac{1}{n}\sum_{k=1}^{n}\frac{|p_c(k) - p_t(k)|}{n} \quad (5)$$

Here, p_c and p_t respectively refer to positions of peer k's computed trust and intrinsic trust values in the ordered list of all peers sorted on the basis of their reputation values.

Overheads. Our reliability metric requires that the network store not only each peer's aggregated trust value, but also a compressed transaction history (in order to compute its reliability value). The transaction history only needs to keep the identity of its past partners and the total number of transactions performed with each partner. We compute this storage overhead as the number of unique transaction partners per peer. Computational and communication overheads for generating our reliability metric are comparable to a traditional reputation system.

We now present the effectiveness of our reliability mechanism in countering collusion. Each data point represents an average of results from at least three randomized runs.

4.3 Resistance to Collusion

Pairwise collusion is the most basic form of collusion, where two peers undertake fake transactions to raise each other's reputation. We vary the percentage of pairwise colluders in the network from 10% to 50% on the X-axis, and plot the transaction success rate on the Y-axis. As shown in Figure 6(a), our reliability-based reputations schemes demonstrate a 80% average success rate, and a 30-40% improvement in network productivity as compared to a pure reputations mechanism. Our mechanism observes little impact with increasing percentage of network colluders. Also, as seen in Figure 6(b), we observe higher accuracy of peer reputations using our reliability scheme despite the increasing amount of collusion in the network.

We also observe that increasing the amount of bootstrap collusion and collusion cost ratios results in a drastic drop in performance of the pure reputations scheme (graphs for bootstrap experiment not included due to brevity considerations). On the other hand, increasing the magnitude of collusion has little to no effect on the success rate of our proposed mechanism. In fact, we observe more accurate results when the amount of pairwise collusion rises in the network, because the inequality in the Lorenz curves for colluders rises sharply when a colluder transacts with even one normal user. Therefore, while these colluders possess

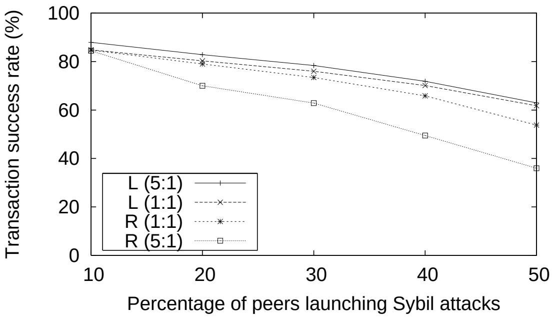

Figure 7. Transaction success rate against Sybil collusion. ***R*** **refers to the pure reputations scheme and** ***L*** **refers to the reliability-based reputations scheme. The ratios represent the collusion cost factors.**

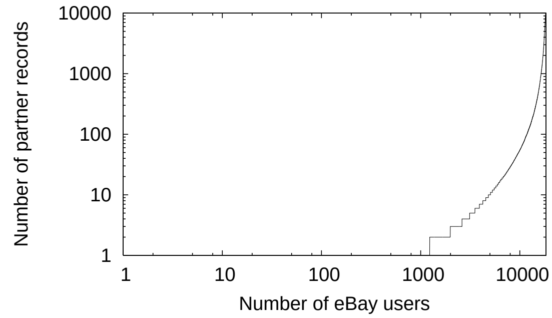

Figure 8. Storage overheads as evaluated by the number of unique transaction partner records maintained for 18,000 eBay sellers.

high reputations, the reliability of their reputations turns out to be really poor.

Next, we evaluate the effectiveness of reliability-based reputations in countering Sybil colluders. A user launching a Sybil attack can establish multiple slave identities and use them to inflate its reputation. We fix the number of bootstrap transactions to 10 and the number of slaves to 5 per attacker. These slaves only behave as service requesters to the master peer and are not a part of the regular peer community. We vary the percentage of Sybil colluders in the network from 10% to 50% on the X-axis, and plot the transaction success rate on the Y-axis. As shown in Figure 7, the pure reputations scheme performs badly with increasing amounts of Sybil attacks and collusion cost ratios in the network. Though we observe a general drop in performance with increasing percentage of users launching these attacks, our mechanism is effective in countering the Sybil attack. We observe a 30% improved success rate even when the Sybils conduct five times as many transactions as normal peers. Similar to our experiment on pairwise collusion, our mechanism observes greater trust computation accuracies as compared to a pure reputations scheme (graph not included for brevity).

A greater number of colluding slaves helps Sybils inflate their reputations with higher reliability scores (as compared to pairwise colluders). However, transacting with even one non-colluder results in a highly disproportionate Lorenz distribution for the Sybil colluder and a sharp drop in its reliability score. Increasing the magnitude of collusion with each of its slaves further aggravates the poor reliability score of the Sybil. A Sybil is challenged to maintain transactions rates per slave comparable to the rates with other non-colluding peers. But this drastically reduces the impact of each colluding partner, resulting in a reputation that more accurately reflects the user's real behavior. We observe similar results for our experiments on the group-based mesh collusion model.

4.4 Storage Overhead

Our objective is to compute additional storage overheads imposed by reliability-based reputations. As part of our performance evaluations, we test our reliability solutions using transaction histories of eBay sellers (Section 5). In Figure 8, we plot the number of unique transaction partners as observed by 18,000 eBay sellers. We observe that 60% of the sellers had less than 100 unique transaction partners, and 85% had less than 1,000 unique transaction partners. We believe that storing an average of 100-1000 transaction records per peer is an acceptable overhead given the improved system productivity we observe from our reliability mechanisms.

5 User Misbehavior in eBay

The previous section evaluated our proposed mechanisms using a simulated peer community. In this section, we evaluate our techniques by applying them on real reputation profiles of eBay users found in the public domain.

eBay (www.ebay.com) is a large and highly successful online marketplace. At any given time, eBay boasts an average of 100 million listings across a diverse community of 233 million users worldwide. The eBay reputation system (a.k.a. the Feedback Forum) enables users to leave feedback following each interaction. An eBay user's reputation is computed as the sum of its lifetime ratings (+1 positive, 0 neutral, -1 negative). Each user can affect another user's feedback score by no more than one point, i.e. multiple transactions with a single partner can only result in a maximum of 1 point change in either direction.

While its reputation system is generally viewed as a success, eBay acknowledges several limitations in its feedback system. First, feedback of all transactions carry the same weight regardless of the transaction value. Second, feedback is almost always positive, possibly driven by users fearful of negative retaliatory feedback. A recent study

Table 1. eBay Transaction Logs

User1	Role	User2	Rating	Date/Time	Type
A	Seller	B	+1	10-12-07 16:10	Open
A	Seller	B	+1	10-12-07 10:14	Open
A	Buyer	D (NR)	-1	10-12-07 10:10	Private
B	Seller	E (PR)	+1	10-10-07 9:40	Private
B	Seller	E (PR)	+1	10-10-07 9:55	Private

shows that sellers receive negative feedback only 1% of the time, indicating that the negative feedback has a much stronger impact in determining a seller's overall reputation [18]. Finally, eBay's feedback score computes a user's reputation as a buyer and a seller within the same metric which makes it hard to interpret its performance as a buyer and seller alone.

The eBay transaction histories provide us with a real data set to test the reliability mechanisms proposed in this paper. In this section, our goal is to investigate the reliability of user reputations on eBay, and possibly identify colluders (and Sybils accounts) by taking a deeper behind-the-scenes look into the transaction histories of real eBay users. While conclusively identifying colluders from transaction histories is impossible, we present strong empirical evidence that suggests user misbehavior.

5.1 The eBay Crawler

We use a web crawler to traverse the graph of transaction histories available at eBay.com. For each user, we obtain detailed transactions of all available transaction histories and parse them for transaction partners, download their feedback profiles and add them to our breadth-first search. We terminated our crawler upon downloading complete transaction histories of approximately 18,000 individual eBay users spanning overall 5 million eBay webpages.

We list the format of the eBay transaction logs in Table 1. *User1* and *User2* refer to the identifiers of the ratee and rater respectively. *Role* refers to the transaction role[1] for which the ratee is getting a feedback rating (as given by *Rating*). *Type* represents the transaction type ("Open" or "Private") depending on whether the item transacted was made public. Finally, a user's status can be either: normal, private, or unregistered. Private users (shown as "PR" in logs) hold their feedback profiles private from the world. Such users are not permitted to sell items on eBay. Unregistered users (shown as "NR" in logs) have canceled their membership or had their membership been suspended by eBay.

5.2 Analysis of eBay Transaction Data

We compute the reliability of each eBay seller's reputation based on the Gini coefficient of its Lorenz curve (Equations 2 and 3). We sort all sellers based on the total number

[1]We focus only on seller histories in our work. Buyers and sellers feedbacks are generally symmetric for each transaction.

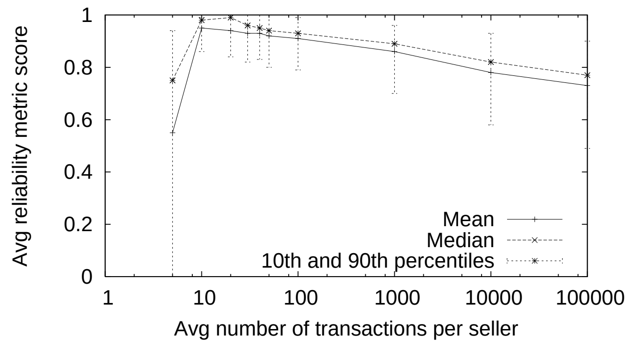

Figure 9. Reliability scores of eBay users calculated using the Gini coefficients.

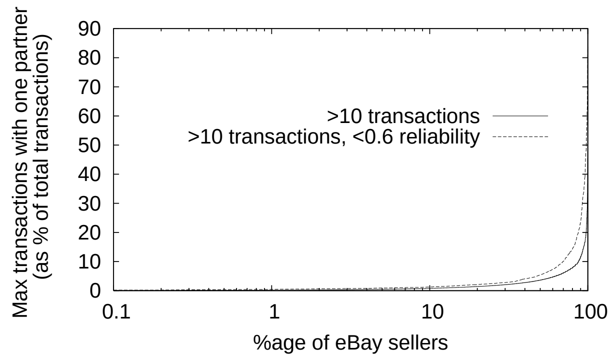

Figure 10. Maximum number of transactions conducted with a single partner calculated as a percent of the seller's total transactions.

of conducted transactions and plot their reliability scores in Figure 9. We make the following observations. First, users with short transaction histories ($<$ 10 transactions) exhibit poor reliability scores. Maximum inequality of a Lorenz distribution occurs when all transactions are conducted with a single user. Almost 25% of the users with less than 10 transactions conducted only one transaction, and therefore have a reliability score of 0. In general, users with less than 10 transactions should find it easier to distribute transactions evenly and generate a high reliability score.

Next, we observe that sellers with 10 to 1000 transactions have high reliability scores ($\sim$61% sellers). Sellers with greater than 1000 transactions have significantly lower reliability scores. Almost half of the sellers with greater than 10000 transactions have scores less than 0.8. This result is counter-intuitive, since we expect users with more transactions to interact more widely, thereby smoothing their Lorenz distribution and improving their score. On the contrary, we observe that many "power users" in eBay demonstrate highly uneven transaction distributions, and consequently, have poor reliability scores. This may be in part due to repeat business from frequent shoppers. Further

Table 2. Detailed eBay Transaction Histories

	Reliability (R)=0.1		R=1
	User 1	User 2	User 3
Total transactions	3599	6051	3793
- As seller	1868	42	3436
Max txns with 1 partner (%)	68.68%	22.5%	0.26%
Total private transactions	3463	4	6
Unique partners as sellers	19	23	3436
Private feedback partners	4	104	0
Unregistered partners	209	102	0
% of positive feedback	100%	99.7%	100%
eBay feedback score	152	707	3609
Partners who left positive	152	708	3609
Partners who left negative	0	2	0
Peak feedback rate per minute	25	19	2

investigation on possible false positives in our data, however, revealed that only 11% of this set ($\sim$ 35 sellers) were registered storefronts on eBay.

We take a closer look at the outliers by examining all sellers with greater than 10 transactions that have extremely low reliability scores (< 0.6). They account for 2.9% of the population. Again, users with less than 10 transactions do not have sufficient transaction history to make an accurate judgment on their reliability. Figure 10 plots, for each seller, the maximum number of transactions conducted with any single partner as a percent of the seller's total transactions. We observe that the outliers have a greater percentage of repeat transactions with one single partner as compared to all eBay sellers. In fact, 5% of these sellers ($\sim$27 sellers) examined had about 35% of their transactions with one single partner. The Lorenz distributions for these sellers are highly disproportionate resulting in their low reliability.

To determine if our filter has found any misbehaving users, we manually examined the online transaction histories of the eBay sellers identified flagged by our reliability filter. We observe some interesting commonalities between these sellers. Table 2 summarizes salient feedback statistics of some of the worst and best eBay sellers as identified by our reliability metric. First, the table shows that peers with poor reliability scores conduct a high percentage of their total transactions with a single user, which suggests reputation inflation via pairwise collusion. We also observe that some of these sellers repeatedly transact with private users. Since private users have very little public information posted, they are ideal candidates in a Sybil-collusion attack.

A look at the feedback logs of these suspicious sellers reveals another trend. These users tend to perform transactions with the same partner at an incredible rate of speed in short time periods. We plot 7 eBay sellers from our filtered user set that hold reliability scores of 0.2 or less, and evaluate their overall feedback rates per minute and the total

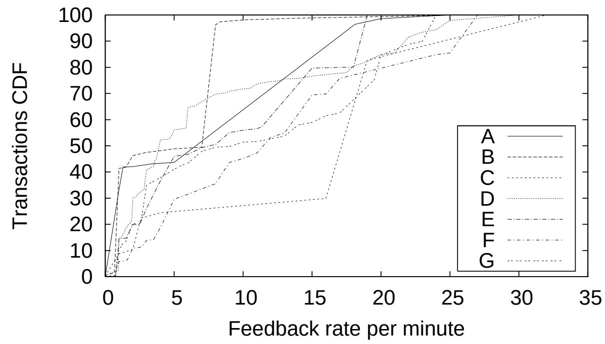

Figure 11. Feedback rates per minute for 7 eBay sellers with reliability scores of less than 0.2.

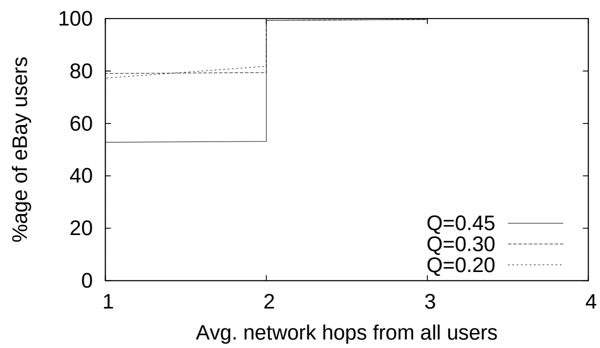

Figure 12. Network distance between suspicious eBay sellers as computed by the Dijkstra's shortest path algorithm.

percent of feedback received at those rates. As seen in Figure 11, these suspicious users receive less than 15% of total feedbacks at rates slower than 2 feedbacks per minute, and about 50% of their feedbacks are received at a rate greater than 8 feedbacks per minute! While an automated script [3] that generates such feedback is possible, suspicious sellers filtered from our metric demonstrate bursts of transactions with more than 19 consecutive feedbacks from the same partner within 1 minute. We are unclear as to the real motive behind these actions, since multiple transactions with a single partner only produces 1 point in the feedback score. We suspect the goal is to receive other incentives such as the title of an *eBay Power User*.

The objective of our last experiment is to determine the existence any malicious cliques in the eBay network. In this experiment, we seed our web crawler with a 0.45 reliability seller and traverse (in breadth-first manner) the graph of all eBay users that also demonstrate a reliability score of 0.45 or less. Our crawl stops when no more users are discovered in the crawler run. We repeat our crawler for reliability scores less than 0.30 and 0.20. Our experiment observed a negligible amount of direct interactions between two suspicious sellers. Therefore, we compute the average distance between two suspicious sellers as follows: we first create a network graph with all suspicious sellers as nodes. For every pair of sellers, we investigate whether the two sellers have any buyers common to them. If one (or more) common buyers exist, we create a network link between the two sellers. We then compute shortest distance between all seller pairs using Dijkstra's shortest path algorithm on the network graph.

Figure 12 plots the average shortest distance between of all sellers with the rest of the network. We observe that the average distance between two sellers decreases with decreasing reliability scores. Almost all sellers with less than 0.45 reliability ($\sim$300 sellers) are fewer than 3 hops away from similar sellers. Almost 80% of the sellers with reliability less than 0.30 ($\sim$23 sellers) are observed to have at least one common buyer between then. This indicates a very clustering among suspicious sellers. This could also in part be because the sellers crawled are typically trading within the same auction category (like electronics, books, etc).

6 Conclusions

Applying reputation systems to large-scale dynamic networks can produce erroneous and misleading values due to collusion between misbehaving users. By leveraging the well-accepted Lorenz curve and Gini coefficient, we provide a reliability metric designed to detect and penalize collusion-like behavior, and encourage peers to interact with diverse groups of peers across the network. Our evaluations find that our metric complements traditional reputations well and quickly isolates a number of users with highly suspicious behavior. Our reputation mechanism helps enable trust and accountability for both application-level networks like eBay, as well as network-level protocols, including message routing, distributed storage and content distribution networks.

References

[1] K. Aberer and Z. Despotovic. Managing trust in a Peer-2-Peer information system. In *Proc. of CIKM*, November 2001.

[2] I. Abraham et al. Practical locality-awareness for large scale information sharing. In *Proc. of IPTPS*, February 2005.

[3] AuctionPixie. `http://www.auctionpixie.co.uk/automatic-feedback.aspx`, 2008.

[4] S. Buchegger and J. L. Boudec. A robust reputation system for P2P and mobile ad-hoc networks. In *Proc. of P2PEcon*, June 2004.

[5] C. Dagum. The Generation and Distribution of Income, the Lorenz Curve and the Gini Ratio. *Economie Applique*, 33:327–367, 1980.

[6] E. Damiani et al. A reputation-based approach for choosing reliable resources in peer-to-peer networks. In *Proc. of CCS*, November 2002.

[7] P. Dewan and P. Dasgupta. Pride: Peer-to-Peer reputation infrastructure for decentralized environments. In *Proc. of WWW*, May 2004.

[8] P. Dewan and P. Dasgupta. Securing reputation data in peer-to-peer networks. In *Proc. of PDCS*, November 2004.

[9] J. Douceur. The sybil attack. In *Proc. of IPTPS*, March 2002.

[10] M. Feldman, K. Lai, I. Stoica, and J. Chuang. Robust incentive techniques for peer-to-peer networks. In *Proc. of EC*, May 2004.

[11] M. Feldman, C. Papadimitriou, J. Chuang, and I. Stoica. Free-riding and whitewashing in peer-to-peer systems. In *Proc. of WEIS*, May 2004.

[12] M. J. Freedman, E. Freudenthal, and D. Mazies. Democratizing content publication with coral. In *Proc. of NSDI*, December 2004.

[13] R. Jurca and B. Faltings. An incentive compatible reputation mechanism. In *Proc. of AAMAS*, June 2003.

[14] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in P2P networks. In *Proc. of WWW*, May 2003.

[15] Q. Lian et al. An empirical study of collusion behavior in the maze p2p file-sharing system. In *Proc. of ICDCS*, June 2007.

[16] M. Lorenz. Methods for Measuring the Concentration of Wealth. *American Statistical Association*, 9:209–219, 1905.

[17] B. C. Ooi, C. Y. Liau, and K.-L. Tan. Managing trust in peer-to-peer systems using reputation-based techniques. In *Proc. of WAIM*, August 2003.

[18] P. Resnick, R. Zeckhauser, J. Swanson, and K. Lockwood. The value of reputation on ebay: A controlled experiment. *Experimental Economics*, 9(2):79–101, June 2006.

[19] S. Rhea et al. Pond: The OceanStore prototype. In *Proc. of FAST*, April 2003.

[20] A. Rowstron and P. Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *Proc. of Middleware*, Nov 2001.

[21] S. Saroiu, P. K. Gummadi, and S. Gribble. A measurement study of peer-to-peer file sharing systems. In *Proc. of MMCN*, January 2002.

[22] I. Stoica et al. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proc. of SIGCOMM*, August 2001.

[23] G. Swamynathan, B. Y. Zhao, and K. C. Almeroth. Decoupling service and feedback trust in a peer-to-peer reputation system. In *Proc. of AEPP*, 2005.

[24] K. Walsh and E. G. Sirer. Experience with an object reputation system for peer-to-peer filesharing. In *Proc. of NSDI*, May 2006.

[25] L. Xiong and L. Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Trans. on Knowledge and Data Engineering*, 16(7), 2004.

[26] M. Yang et al. Deployment of a large-scale peer-to-peer social network. In *Proc. of WORLDS*, December 2004.

[27] H. Yu et al. Sybilguard: Defending against sybil attacks via social networks. In *Proc. of SIGCOMM*, September 2006.

[28] H. Yu, J. Rexford, and E. Felten. A distributed reputation approach to cooperative internet routing protection. In *Proc. of NPSec*, November 2005.

[29] H. Zhang et al. Making eigenvector-based reputation systems robust to collusion. In *Proc. of WAW*, October 2004.

[30] B. Y. Zhao et al. Tapestry: A global-scale overlay for rapid service deployment. *IEEE JSAC*, 22(1), January 2004.