Game Theoretical Analysis of Incentives for Large-scale, Fully Decentralized Collaboration Networks

Thomas Bocek*, Michael Shann*, David Hausheer* and Burkhard Stiller*†
*Communication Systems Group
Department of Informatics IFI, University of Zurich
Email: bocek|hausheer|stiller@ifi.uzh.ch, mikeshann@access.uzh.ch
†Computer Engineering and Networks Laboratory
ETH Zurich
Email: stiller@tik.ee.ethz.ch

***Abstract*—One of the key challenges in peer-to-peer networks is the design of incentives to encourage peers to share their resources. Incentives are necessary in order to exploit the full potential of these systems. The tit-for-tat incentive scheme, as used in BitTorrent for example, has proven to be a successful approach in P2P file sharing systems, where peers have direct relations and share the same kind of resources. However, in P2P systems where different kind of resources are shared between peers with non-direct relations, the design of incentives remains a challenge. In this paper, a large-scale, fully decentralized P2P collaboration network is shown, where peers share not only bandwidth and storage space, but also contribute by editing articles and voting for or against changes. A new incentive scheme is proposed which supports non-direct relations and provides incentives for sharing different kind of resources. The incentive scheme is based on a reputation system that assigns a reputation value to every peer reflecting its previous behavior in the network. Based on this value, the service level is differentiated, i.e. the higher a peer's reputation the better the quality of service it can get from the network. The service differentiation has been analyzed and simulated with rational, irrational and altruistic peers based on game theory concepts.**

I. Introduction

Peer-to-peer (P2P) networks have received much attention in the past because they have many advantages over client-server architectures, including fault tolerance, higher scalability, and less maintenance costs.

However, one of the drawbacks of P2P networks is that participants tend not to share their resources, which can result in a serious loss of performance [1], [8]. Thus, in order to encourage peers to share their resources there needs to be an incentive scheme that rewards cooperation and punishes free-riding.

BitTorrent [4] is a prominent example of a P2P file-sharing system that includes a solid incentive scheme called tit-for-tat (TFT). TFT provides incentives to share resources for peers with direct relations and resources of same kind. Two peers are in a direct relation if both peers demand and supply resources of the same kind. The TFT scheme is useful for applications with many direct relations. This is typically the case for P2P file-sharing systems, especially when sharing popular files.

In a P2P collaboration network, however, the peers' relations are often non-direct and resources are of different kind. Besides contributing bandwidth and storage space, in a P2P collaboration network a peer also contributes with editing of documents and voting for or against changes of documents. The voting mechanism ensures a high quality of those documents [3], [17]. In such a system, TFT cannot be applied as it does not support non-direct relations.

Therefore, this paper proposes a new incentive scheme for large-scale, fully decentralized P2P collaboration networks which overcomes the shortcomings of TFT. The proposed incentive scheme is based on a reputation mechanism, which assigns every peer a certain reputation value that reflects the peer's previous behavior in the network. The services a peer can get from other peers are differentiated based on that value: the higher a peer's reputation, the higher the quality of service the peer can get.

In general, a reputation mechanism consists of the following three parts:

1) Definition of metrics to calculate a peer's reputation
2) Secure and efficient propagation of reputation values
3) Differentiation of services based on reputation values

A lot of work has already been done on 2., i.e. the problem of reputation management [9], [5], [13]. In contrast to the issue of propagation of reputation values, the problem of defining a metric for calculating a peer's reputation has not received much attention. Most approaches assume a simple reputation metric, where the reputation of a peer is based on a single aspect, *e.g.*, the number of files uploaded. Consequently, the focus of this paper will be on the definition of reputation metrics and on the service differentiation. The existence of a mechanism to safely propagate reputation values in a P2P network is assumed.

The proposed reputation metric and service differentiation is analyzed and simulated with rational, irrational and altruistic peers using game theoretical approaches. The simulation model is a collaboration network with resources and services such as sharing documents, editing documents, and voting for documents. Results show that the amount of shared resources can be increased by at least 8-10%. Additionally, this paper shows that rational peers behave according to the majority. If the majority behaves irrationally, the rational peers will adapt to this.

The rest of the paper is structured as follows: First, Section 2 discusses related work, before the design of the actual reputation-based incentive scheme is presented in Section 3. Afterwards, the simulation model is outlined in Section 4. Section 5 presents the results from the simulation and analysis of the effectiveness of the scheme. Finally, Section 6 draws conclusions.

II. Related Work

Related work in the area of reputation-based incentives for P2P networks is addressed as follows. Game theory provides the necessary basis to analyze incentives in general. Then, two categories of incentive schemes are outlined. Finally, a brief overview of approaches for shared history reputation management is given.

A. Game Theory

Game Theory suggests that people behave *rationally*, i.e. given a set of possible actions they will try to take the one that maximizes their utility, where the utility is the difference between the benefit and the costs of the action [6]. For file sharing in a P2P network, this means that if a user can choose between sharing or not sharing a file he will tend not to share it, i.e. free-ride the system, because that inflicts less cost upon him. However, this model does not fully reflect reality, as there are typically peers who share without seeking a benefit. This shortcoming is addressed by introducing the following standard *behavior types*: altruistic, rational, and irrational [15]. An altruistic user contributes to the system without weighing benefits against costs, whereas an irrational user acts unpredictably and anti-socially. Examples of irrational behavior are, *e.g.*, the spreading of viruses or online vandalism. A repeated play of the Prisoner's Dilemma seems to be an appropriate model of interaction among users in a P2P network [5]. A very effective strategy [2] to play the repeated Prisoner's Dilemma is Tit-for-tat, as for example BitTorrent [4] implements it.

Ma et al. show game theoretic approaches to provide incentives and service differentiation [11]. The authors designed a protocol with a bidding mechanism for resource distribution. Contribution values are stored in an auditing authority. Their mechanism achieves Pareto-optimal allocation results and can adapt to node joins and node failures.

B. Categories of Incentive Schemes

Incentive schemes can be divided into two categories: trade based and trust based schemes [12]. Trade based schemes are based on some kind of currency, *e.g.*, micropayments, which can be used by peers in exchange for resources shared with other peers. Every file, for instance, has a price that has to be paid before downloading. Trust based incentive schemes, on the other hand, manage a reputation value of each peer which reflects the peer's previous behavior in the system. A peer with a high reputation is likely to get better services than a peer with a low reputation.

1) Trade based Schemes: Trade based schemes such as Offline Karma [7] are very efficient from an economic point of view. Every peer gets exactly as much from the system as it is willing to contribute. The main drawback of these schemes is that in order to manage some sort electronic currency, they either require a central authority or they produce a lot of overhead in terms of communication costs, both of which is not desirable.

2) Trust based Schemes: Trust based schemes can be divided into private or shared histories [5]. With a private history every peer keeps track of the behavior of other peers in direct relation and adapts its policy. With a shared history the actions of all peers are known, i.e. a peer can adapt its policy to any other peer even without direct relation.

C. Reputation Systems

Service differentiation is designed in such a way that users will strive to get a high reputation value for a high Quality-of-Service (QoS). In order to get a high reputation, users should be forced to social behavior, i.e. behavior beneficial to other peers. Before service differentiation can take place, the peer's reputation value has to be computed and made accessible to all other users. The issue of propagating reputation values safely, i.e. by avoiding any collusion and false reports, is a challenge. Several proposals have already been made to tackle these problems. The EigenTrust algorithm [9] is an elegant and efficient way of computing global trust values. It works similar to the PageRank algorithm used by Google [13]. In the EigenTrust approach, the global trust value of peer k is the k-th component of the left principal eigenvector of the trust matrix $C = (c_{ij})$, where c_{ij} for peer i is the local reputation of peer j. Unfortunately, EigenTrust is not safe against collusion. For example, peers can boost their reputation score by simply uploading some files to a highly reputable peer [10]. Additional mechanisms are needed to prevent these kinds of attacks.

Another approach is the Maximum Flow (MaxFlow) algorithm [5], which determines the maximum 'flow' that can be achieved between a source node and a target node in a directed graph, given some capacity constraints for every edge [14]. For a P2P network, the source and target node are interpreted as users, and the constraints refer to the local trust value a peer assigns to its neighboring peers. Then the maximum flow is the maximum reputation the source node can assign to the target node without violating reputation constraints.

III. Reputation Design

This section defines how the reputation value is calculated from the actions of a peer. First, the reputation function and the contribution value functions are provided. Then, the service differentiation functions and the utility functions are shown. The goal of these mechanisms is to stimulate peers to actively participate in a P2P network. Section IV defines a model of a P2P network to analyze and measure the efficiency of those mechanisms.

A. Reputation Function

First, general properties of reputation values are shown before proceeding to the concrete definition of the reputation function. Reputation values are based on how much a peer contributes to the P2P network. This contribution value is defined as $C(action_1, action_2, ..., action_n) \geq 0$, where $action_i$ $i = 1, 2, ..., n$ are its actions undertaken. The more a peer contributes, the higher is its contribution value and, thus, its reputation value.

If a peer joins the network its initial reputation value R_{min} has to be larger than 0. Otherwise, the peer could not download anything from other rational peers, because resource consumption such as downloading depends on the reputation of a peer (see Section III-C). On the other hand, a high R_{min} provides incentives for whitewashing the identity. The reputation value also needs to have a certain maximum value $R_{max} = 1$. Finally, the reputation value should increase quite fast at the beginning, in order to motivate newcomers to contribute to the system.

To meet these requirements, the reputation value R has been defined as a monotonically increasing function in the contribution value C, $R : \Re_{\geq 0} \to [R_{min}, 1]$. As a representation of R a logistic function

$$R(C) = \frac{1}{1 + g \cdot \exp(-\beta \cdot C)}$$

is used, which satisfies all the above properties. Figure 1 shows this function for *e.g.*, $g = 19$ and different values of β.

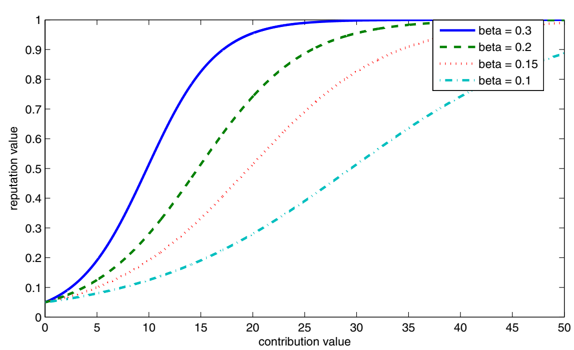

Figure 1. Reputation function: $g = 19$, β

B. Contribution Value

While the number of shared articles and the amount of shared bandwidth is easy to measure, the effort involved in editing and voting is difficult to quantify. Therefore, there is no simple answer to questions like storing how many articles is equal to editing an article. To circumvent this problem, two different contribution values are introduced: one for sharing articles and bandwidth and another one for voting and editing. The justification for this is that sharing articles and bandwidth are comparable to each other, as are voting and editing. As a consequence, each user has two reputation values:

- $R_S(C_S)$, the reputation for sharing articles and bandwidth
- $R_E(C_E)$, the reputation for voting and editing

From a global perspective, storing and editing articles as well as sharing bandwidth and voting are all desirable actions. Therefore, the contribution value should increase, if one of these actions are undertaken. To be more precise, editing articles and voting are only desired if they are done constructively, i.e. if the intention is to improve the quality of the articles.

The contribution values are defined as the weighted sum of their respective desirable actions.

1) Contribution Value for Sharing Bandwidth and Storage: First, the contribution value for article and bandwidth sharing is

$$C_S(a, b) = \alpha_S \cdot S_{articles} + \beta_S \cdot S_{bandwidth} - d_S$$

where

- $S_{articles}$ are the actually shared articles
- $S_{bandwidth}$ is the actually shared bandwidth
- $\alpha_S \in \Re_{>0}$, $\beta_S \in \Re_{>0}$ are constants that are used to weight the different actions. For instance, $\alpha_S = 1$ and $\beta_S = 2$ means that sharing bandwidth is twice as valuable as offering articles for download.
- $d_S \in \Re_{>0}$ is a constant, the decay term. The decay term works as follows: if a peer is inactive for some time its contribution value will decrease.

2) Contribution Value for Editing and Voting: Similarly, the contribution value for editing and voting is defined as

$$C_E(v, e) = \alpha_E \cdot S_{votes} + \beta_E \cdot S_{edits} - d_E$$

where

- S_{votes} are the successful votes. A vote is successful, if and only if it is cast with the majority
- S_{edits} are the accepted edits. An edit is accepted, if and only if a majority votes for the edit
- $\alpha_E \in \Re_{>0}$, $\beta_E \in \Re_{>0}$ are constants similar to α_S and β_S.
- $d_E \in \Re_{>0}$ is a decay term analog to d_S

C. Service Differentiation

Service differentiation is discussed using the following services. Downloading an article, editing an article, and casting a vote on an edit.

1) Downloading: If several peers want to download a file from the same source, they compete for the source's upload bandwidth. With the incentive mechanism, a peer i gets the amount of bandwidth equal to the ratio of its sharing reputation R_S compared to the sum of the sharing reputation of all peers that download from the same source. More formally, let D_j be the set of peers that download from peer j. Then, for each peer $i \in D_j$ the percentage of bandwidth B_i it gets from peer j's upload bandwidth is

$$B_i = \frac{R_S^i}{\sum_{k \in D_j} R_S^k}$$

2) Voting: In order to get a set of constructive and capable voters, voting privileges are limited to those users that take voting seriously. It is assumed that if a peer has previously edited an article, it has knowledge of the subject and little interest in disruptive behavior, at least concerning its own article. Therefore, only successful editors of an article will get the right to vote on changes of that article.

There are two additional mechanisms to provide incentives in the voting process: weighted voting and punishment of malicious voters. Weighted voting means that a peer's voice counts in proportion to its voting and editing reputation R_E. If V is the set of voters on an edit, then each peer $i \in V$ has the voting power

$$v_i = \frac{R_E^i}{\sum_{k \in V} R_E^k}$$

Punishment of malicious voters works as follows: if the number of a peer's unsuccessful votes, i.e. votes against the majority, exceeds a certain threshold it will lose its voting rights. To get any new rights, the peer has to contribute constructive edits first.

3) Editing: In order to make vandalism more difficult, an initial cost for the editing is incurred. Before a peer gets the ability to edit articles it must have contributed some resources to the network, i.e. its sharing reputation must be above a certain threshold: $R_S \geq \theta > R_S^{min}$.

Moreover, peers who have written many articles need less consent on their edits because generally they are expected to make good edits. On the other hand, peers with few edits need more votes for a successful edit. Therefore, the majority M of a vote is inversely proportional to the editor's reputation.

Punishment of malicious editors works as follows: if a peer has too many declined edits it will lose its editing right. This is done by setting its sharing reputation to the minimum value: $R_S = R_S^{min}$. In addition, the editing reputation R_E drops to the minimum value as well: $R_E = R_E^{min}$.

D. Utility Function

The utility functions for sharing resources, editing, and voting are defined as follows. For simplicity, all peers are assumed to have the same download and upload bandwidth, which is normalized to 1. Moreover, the file size is set to 1 for all files. Thus, if a peer uploads a file its upload bandwidth is always fully used.

1) Utility for Sharing Articles and Bandwidth: The utility for sharing articles and bandwidth is defined as

$$U_S = \alpha \cdot UP_{source} \cdot B - \beta \cdot DS_{articles} - \gamma \cdot UP_{own}$$

where

- UP_{source} is the source's shared upload bandwidth
- B is the percentage of the download bandwidth of the peer as defined in Section III-C1
- $DS_{articles}$ is the percentage of disk space used for sharing articles
- UP_{own} is the percentage of upload bandwidth shared by the peer itself
- $\alpha, \beta, \gamma \in \Re$ are constants to reflect the benefit of downloading and the cost of sharing

The utility U_S can take positive and negative values, depending on the peer's sharing and downloading behavior.

2) Utility for Editing and Voting: Similarly, the utility of editing and voting is defined as

$$U_E = \delta \cdot E_{succ} + \epsilon \cdot V_{succ}$$

where

- E is the number of successful edits
- V is the number of successful votes
- $\delta, \epsilon \in \Re$ are modifiers analog to α, β, γ

The cost of editing and voting are not considered in the formula because both of these cannot be explained rationally. There must be an altruistic motivation for them.

IV. Simulation Model

The formal simulation model to test the effectiveness of the incentive mechanism described in the previous section, is outlined. The model is based on the assumptions of Game Theory that have been described in Section 2.

In the model, time is discretized. At every time step, a peer downloads an article from another peer with probability $P = \frac{1}{N_S}$, where N_S is the number of peers that offer any files for download. At the same time, the peer chooses how many files and how much bandwidth it wants to share. It is also possible to edit an article and to vote on any changes.

In the simulation, every peer is represented by a self-learning agent that will try to maximize its benefit by exploring different strategies. The learning algorithm is Q-Learning, an efficient temporal-difference reinforcement learning algorithm [16]. Before proceeding to the simulation itself, the concept of Q-Learning is briefly described.

A. Q-Learning

In reinforcement learning, and particularly in Q-Learning, there is an agent interacting with its environment over a set of actions. For each action it gets a feedback or reward from the environment, and over time it learns what actions are best in a given situation to maximize the utility. More formally, time is divided into discrete steps $t = 1, 2, 3, ...$. At each time step t, the environment is in a state $s_t \in S$, where S is the set of possible states, and the agent chooses an action $a_t \in A(s_t)$, where $A(s_t)$ is the set of possible actions in state s_t. It gets a reward r_t, and the state of the environment changes to s_{t+1}. The agent's goal is to maximize its expected reward

$$R = r_0 + \gamma \cdot r_1 + \gamma^2 \cdot r_2 + \gamma^3 \cdot r_3 + ... = \sum_{t=0}^{\infty} \gamma^t \cdot r_t$$

where $0 \leq \gamma \leq 1$ is a discounting factor that decides how much the agent considers future payoffs.

Q-Learning solves this task by assigning every state-action pair $(s, a) \in S \times A$ the Q-Value $Q(s, a)$ that equals the expected reward if action a is undertaken in state s and a fixed strategy is followed thereafter:

$$Q(s, a) \leftarrow (1 - \alpha) Q(s, a) + \alpha \left(r + \gamma \max_{b \in A} Q(s', b) \right)$$

where $\alpha \in \Re$ is the learning rate and s' is the state that follows from taking action a in state s.

To solve the exploration-exploitation problem, the agent chooses an action probabilistically over a Boltzmann distribution (which is a probability distribution originally used in Thermodynamics). In general, an action with a high Q-Value should have a higher probability to be chosen. In each round, the probability to take action a in state s is:

$$p_s(a) = \frac{\exp\left(\frac{Q(s,a)}{T}\right)}{\sum_{b \in A} \exp\left(\frac{Q(s,b)}{T}\right)}$$

$T \in \Re$, the 'temperature', controls the amount of exploration. A high T makes the agent choose an action almost uniformly at random, whereas a low T makes high Q-Values more probable. Figure 2 shows two Boltzmann distributions for the values $x = 1, 2, ...10$ and the temperatures $T = 2$ and $T = 1.000$.

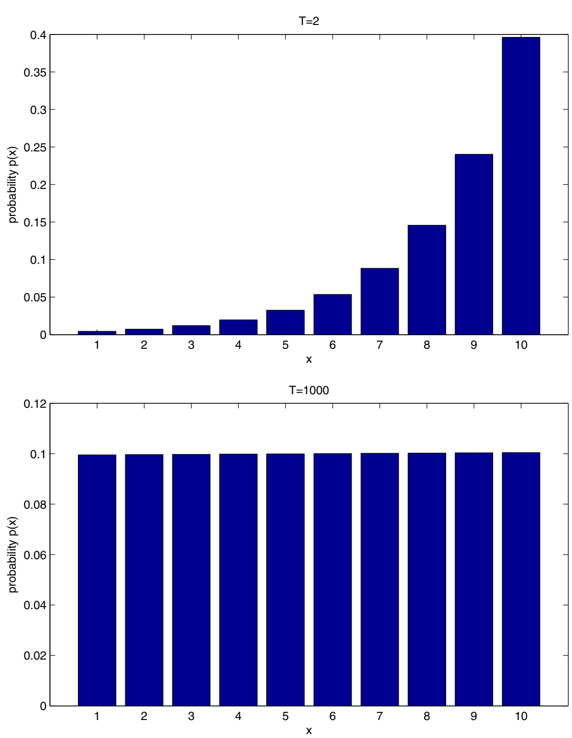

Figure 2. A Boltzmann distribution for values $x = 1, 2, ..., 10$, $T = 2$ (top) and $T = 1000$ (bottom)

B. Simulation Setting

In the simulation, there is a training phase of 10.000 time steps in which T is set to the highest possible floating-point value. This grants that the agents will explore all actions with equal probability and that no agent will have a degenerated Q-Matrix. After that, the reputation values are reset but the agents keep their Q-Matrices. In this phase, T is set to 1 so that the agents choose their actions weighted towards the ones with the highest Q-Values.

For the simulation, there are 10 states, where each state consists simply of the agent's reputation. Let $R_{min} = 0.05$. Then, each state represents $\frac{1}{10}$ of the reputation interval $[0.05, 1]$. The network consists of 100 agents. With regard to sharing, an agent can choose from three different participation levels for each resource: 0%, 50% or 100% of their bandwidth; and 0, 50 or 100 files. If an agent is interested in editing and voting, it can do it either constructively or destructively. The following convention regarding user types is used: rational peers always try to maximize their benefit, irrational ones are always free-riders with regard to sharing as well as destructive editors and voters. Altruistic peers always share the most they can and perform only constructive edits and votes. In the simulation, the occurrence of each user type is varied from $10 - 100\%$ while the other two types each share half of the difference to 100%. So for example, in the first run there are 10% rational and 45% altruistic and irrational users. Then 20% rational and 40% altruistic and irrational users, and so on.

V. Results

The amount of shared resources with incentive and without incentive mechanisms are compared. Moreover, the performance under the influence of different mixtures of user types is evaluated.

The metrics of interest are the percentage of shared files and bandwidth per user and especially per rational user. Furthermore, the ratio of constructive to destructive edits and the percentage of accepted constructive edits are measured.

A. Effectiveness with Rational Peers

In this result set, only rational peers are considered. Figure 3 shows the amount of shared articles and bandwidth with and without the incentive scheme. If the incentive scheme is applied, the amount of sharing is higher: the peers share approximately 8% more articles and approximately 11% more bandwidth.

The results show that even though the agents share more resources if the scheme is applied, the difference is not as big as one might expect. An explanation for this might be that sharing more resources does not pay off and, therefore, a peer chooses to stay at a low reputation level. This is due to the fact that the percentage of upload bandwidth a peer gets, increases linearly with the reputation value, while the reputation function itself flattens very quickly after the inflection point. This means that after this point the agents have to spend much more resources than they can get back from downloading, which results in the above behavior.

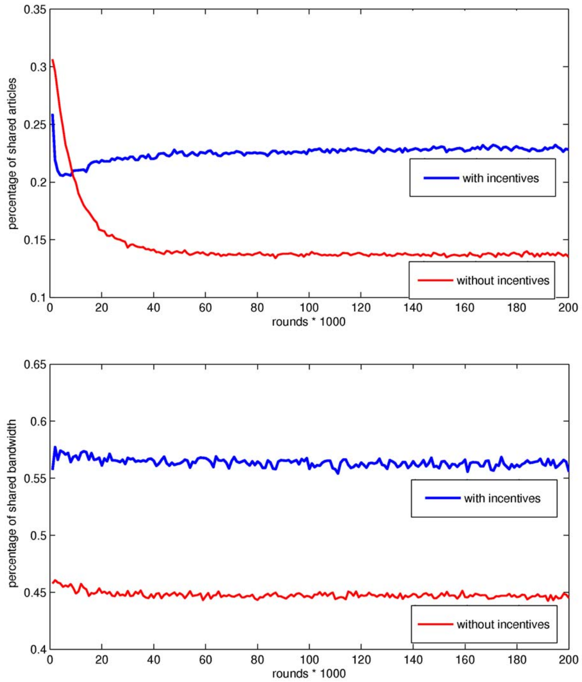

Figure 3. Amount of shared articles (top) and bandwidth (bottom) with rational peers

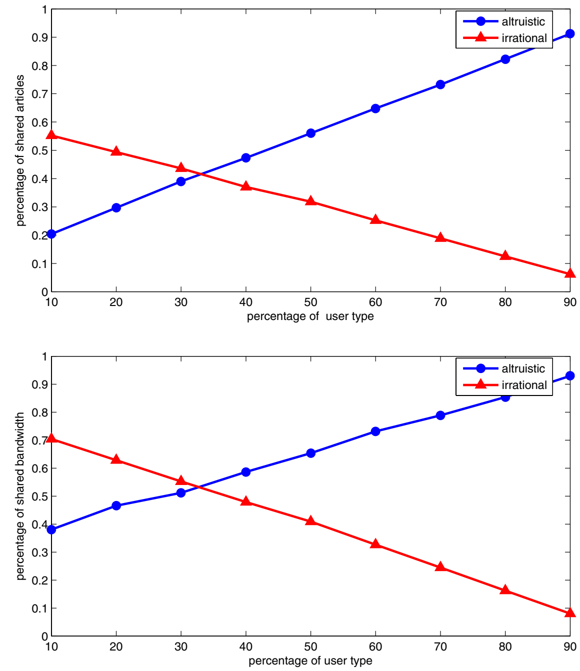

Figure 4. Amount of shared articles (top) and bandwidth (bottom) per peer with varying degrees of altruistic (circle) and irrational (triangle) peers

B. Influence of Different User Types

In this result set, different mixtures of behavior types are considered. As expected, the overall network performance increases in the number of altruistic agents and degrades in the number of irrational ones, as can be seen in Figure 4. If sharing articles and bandwidth is considered, this effect is nearly linear, that is, the behavior of rational agents does not seem to be affected by varying degrees of altruistic and irrational agents (see Figure 5).

Rational agents share their resources even if many irrational agents are present. On the other hand, altruistic agents do not boost resource sharing. One could have expected that if there are many altruistic agents, they exert some pressure on rational peers because altruistic agents generally have high reputation values and, therefore, occupy larger parts of the upload bandwidth than their rational counterparts. However, this effect was not observed.

With regard to editing, the simulations show a different picture. Rational peers do their edits in such a way that their article will be accepted by the community. Their voting behavior is similar, i.e. they will try to vote with the majority. As a consequence, the general tendency of the network - if peers behave constructively or destructively - depends strongly on the number of altruistic and irrational peers (see Figure 7). If the network consists solely of rational agents the outcome is completely random (see Figure 6). This is because initially, the rational agents try constructive and destructive voting and editing behavior at random. They get their reward depending on how many other agents exhibit the same behavior like

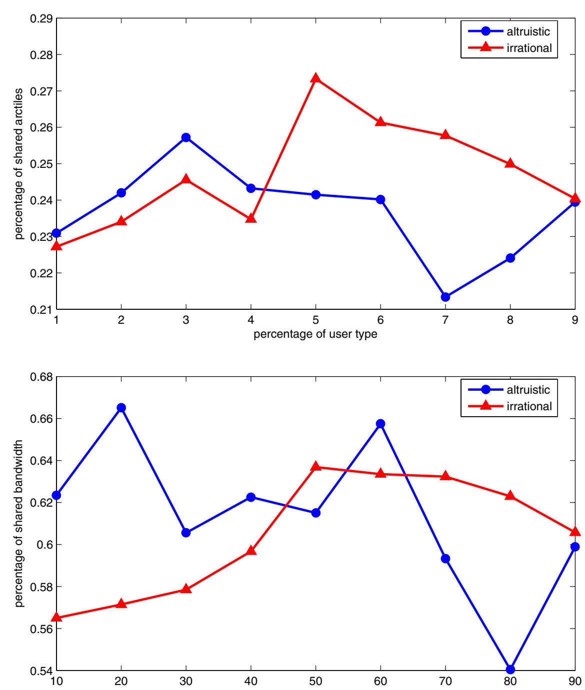

Figure 5. Amount of shared articles (top) and bandwidth (bottom) per rational peer with varying degrees of altruistic (circle) and irrational (triangle) peers

themselves. For example, if 60% of the agents have selected a destructive voting behavior, the chance to succeed with destructive voting behavior is bigger than with constructive behavior. This way, the agents may learn that constructive behavior does not pay off.

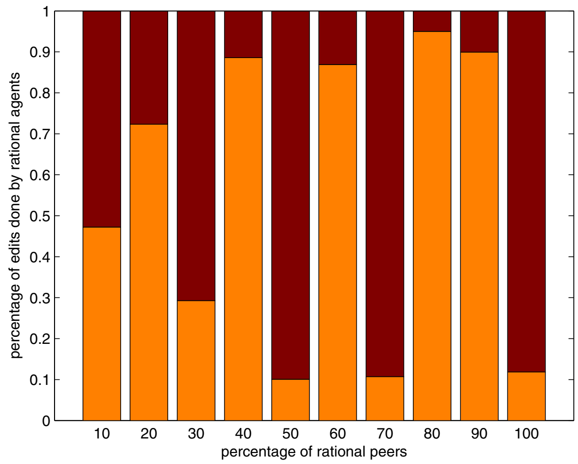

Figure 6. Percentage of constructive (orange/grey bar) and destructive (brown/dark bar) edits by rational agents if number of altruistic and irrational agents are equal

However, if there are more altruistic than irrational agents the simulations show a clear result: the rational agents learn to behave constructively, thereby creating an even greater majority of 'good' agents so that eventually almost all good edits are accepted whereas the bad ones are declined. An analog conclusion holds if there are more irrational than altruistic agents.

VI. Summary, Conclusions, and Future Work

In this paper, a reputation-based incentive scheme for large-scale, fully decentralized peer-to-peer collaboration networks was designed. Its main goals are i) to encourage the participants of the network to share their resources such as bandwidth and storage space; and ii) to ensure a certain quality of the documents that are shared. The main idea of the scheme is to assign a reputation value to every peer reflecting its previous behavior. Based on this reputation value, the quality of service is determined: the higher a peer's reputation the higher its quality of service.

The scheme is vulnerable to the presence of too many malicious peers with regard to editing. This means that in order for the scheme to work as intended, initially there must be more constructive than destructive peers. However, this requirement should be met in a real network, because the first users, *e.g.*, the founders of the network, are expected to have a strong interest to ensure the quality of the network.

In a simulation with self-learning agents, the scheme was shown to be very robust, but moderately effective with regard to resource sharing. The reputation function has a great influence on how much resources are shared. Thus, future work

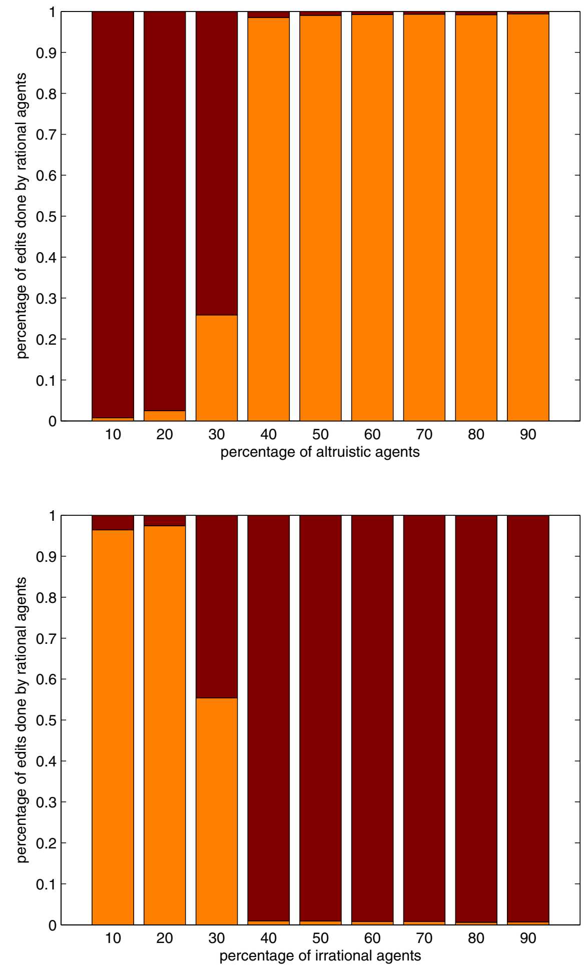

Figure 7. Percentage of constructive (orange/grey bar) and destructive (brown/dark bar) edits by rational agents under influence of varying number of altruistic (top) and irrational (bottom) agents

will investigate new and existing reputation function in order to maximize sharing of resources. In a next step, this new incentive scheme will be adapted and implemented in a P2P network for further simulations.

Acknowledgment

This work has been performed partially in the framework of the EU IST Project EC-GIN (FP6-2006-IST-045256) as well as of the EU IST NoE EMANICS (FP6-2004-IST-026854).

References

[1] E. Adar and B. A. Huberman, "Free Riding on Gnutella," *First Monday, Internet Journal*, vol. 5, no. 10, October 2000.

[2] R. M. Axelrod, *The Evolution of Cooperation*. Basic Books, 1984.

[3] T. Bocek and B. Stiller, "Peer-to-peer large-scale collaborative storage networks," in *Adaptive Infrastructure, Management and Security (AIMS 2007)*, Oslo, Norway, 2007.

[4] B. Cohen, "Incentives Build Robustness in BitTorrent," in Workshop on Economics of Peer-to-Peer Systems, Berkeley, CA, USA, June 2003.

[5] M. Feldman, K. Lai, I. Stoica, and J. Chuang, "Robust Incentive Techniques for Peer-to-Peer Networks," in *EC '04: Proceedings of the 5th ACM conference on Electronic commerce*. New York, NY, USA: ACM Press, 2004, pp. 102–111.

[6] R. H. Frank, *Microeconomics and Behavior*. McGraw-Hill/Irwin, February 2005.
[7] F. D. Garcia and J.-H. Hoepman, "Off-line Karma: A Decentralized Currency for Peer-to-peer and Grid Applications," in *3th Applied Cryptography and Network Security (ACNS 2005)*, ser. Lecture Notes in Computer Science, A. K. J. Ioannidis and M. Yung, Eds., vol. 3531. New York, NY, USA: Springer Verlag, June 7–10 2005, pp. 364–377.
[8] D. Hughes, G. Coulson, and J. Walkerdine, "Free Riding on Gnutella Revisited: The Bell Tolls?" *Distributed Systems Online, IEEE*, vol. 6, no. 6, p. 1, June 2005.
[9] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The Eigentrust algorithm for reputation management in P2P networks," in *WWW '03: Proceedings of the 12th international conference on World Wide Web*. New York, NY, USA: ACM Press, 2003, pp. 640–651.
[10] Q. Lian, Y. Peng, M. Yang, Z. Zhang, Y. Dai, and X. Li, "Robust Incentives via Multi-level Tit-for-tat," in *5th Int. Workshop on Peerto-Peer Systems (IPTPS)*, Santa Barbara, CA, USA, February 2006.
[11] R. T. B. Ma, S. C. M. Lee, J. C. S. Lui, and D. K. Y. Yau, "A game theoretic approach to provide incentive and service differentiation in P2P networks," in *SIGMETRICS '04/Performance '04: Proceedings of the joint international conference on Measurement and modeling of computer systems*. New York, NY, USA: ACM, 2004, pp. 189–198.
[12] P. Obreiter and J. Nimis, "A Taxonomy of Incentive Patterns - The Design Space of Incentives for Cooperation," in *Proceedings of the Second International Workshop on Agents and Peer-to-Peer Computing (AP2PC'03)*. Melbourne, Australia: Springer LNCS 2872, July 2003.
[13] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford Digital Library Technologies Project, Tech. Rep., January 1998.
[14] R. Sedgewick and M. Schidlowsky, *Algorithms in Java, Part 5: Graph Algorithms*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2003.
[15] J. Shneidman and D. C. Parkes, "Rationality and Self-Interest in Peer to Peer Networks," in *2nd Int. Workshop on Peer-to-Peer Systems (IPTPS'03)*, Berkeley, CA, USA, February 2003.
[16] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning)*. The MIT Press, March 1998.
[17] L. Svensson, "Decentralized Secure and Incentive-compatible Voting In P2P Networks," Master's thesis, Communication Systems Group, IFI, University of Zurich, March 2007.