An Enhanced N-Way Exchange-Based Incentive Scheme for P2P File Sharing (Short Paper)*

Lingli Deng[1,2], Yeping He[1], Ziyao Xu[1,2], and Chunyang Yuan[1,2]

[1] Institute of Software, Chinese Academy of Sciences, Beijing 100080, PRC
[2] Graduate School of the Chinese Academy of Sciences, Beijing 100049, PRC
{denglingli, yphe, ccxu}@ercist.iscas.ac.cn, chunyang03@ios.cn

Abstract. Cooperation between participants is essential to P2P applications' viability. Due to obscure possibility to match peers' needs and supplies in pairs, the widely used pair-wise exchange-based incentive schemes perform poorly. The N-way exchange-based incentive scheme enlarges the matching possibility by introducing n person exchanges. But some old problems remain and some new ones emerge with the N-way design. In this paper we present an enhanced n-way exchange-based incentive scheme for P2P file sharing systems. By distributing extra tasks to all the peers involved in an n-way exchange, the proposed scheme eliminates prohibitive computation and communication cost on the cooperators, resulting in greater efficiency, effectiveness, and security.

Keywords: Peer-to-Peer Security, Incentives for Cooperation, Incentives for Enforcement, Exchange-Based Incentive Schemes, Fake Object Attacks.

1 Introduction

As other distributed systems, peer-to-peer file sharing systems rely on the cooperation of autonomous self-interested peers to achieve their global goals. The symmetric relationship between these so-called servents (i.e. the combination of server and client) dictates the traditional cooperative mode, in which the server are motivated and trusted to provide the expected service to its clients, is not applicable any more. Instead, there is an inherent tension between individual rationality and collective welfare[1]. "Free-riders", users who attempt to use the resources of others without contributing their own, manifest to be a severe problem: it was found in 2000, that in the Gnutella, approximately 70% of peers were free-riding[2]; five years later, the number has increased to 85%[3]. Similar problems caused by the lack of incentive for cooperation pervade nearly all the popular P2P file-sharing networks[4][5][6]. And the characteristics of P2P

* Supported by National High-Tech Research and Development Program of China (863) under Grant No. 2002AA141080; National Natural Science Foundation of China under Grant No.60073022 and 60373054; Graduate Student Innovation Grant of Chinese Academy of Sciences.

P. Ning, S. Qing, and N. Li (Eds.): ICICS 2006, LNCS 4307, pp. 322–331, 2006.

systems (e.g. large population and high turnover, asymmetry of interest, collusion, zero-cost identity, and traitors[7]) call for an effective, efficient and secure incentive mechanism to alleviate or overcome the free-rider problem.

The term "incentive" can be seen as remuneration for peers' transactional behaviors[8]. According to remuneration type, existing incentive schemes fall into three major categories: exchange-based, reputation-based and monetary schemes. Exchange-based incentive scheme possesses several elegant features, making it the most desirable alternative in a distributed environment without a centralized authority[9]: It's simple, no centralized authority or dedicated infrastructure is needed like the monetary schemes do; and it's safer than the reputation-based schemes for its immunity to collusion through direct object exchanging. However, primitive pair-wise exchange schemes with the simple tit-for-tat strategy[10][11], are performing unsatisfactorily[12], due to the difficulty for the serving peer to predict which one he is serving to would be serving him in the future. Thus he has to be unnecessarily generous in giving. And this extra generosity can be exploited by tactful free-riders. The problem can be conquered by extending the pair-wise exchange to an n-way pattern[13], where every cooperator gets from its directly upstream peer and gives to its directly downstream peer along a n-way exchange ring. However, the burden of locating and verifying candidate exchange rings, which is put entirely onto the responding peers' shoulders, weakens the incentive for these serving but also rational peers to enforce the n-way scheme, and ultimately erodes its effectiveness in suppressing free-riding. Moreover, in exchange-based schemes there's no explicit incentive for rich peers not currently in need to cooperate.

In this paper we present an enhanced n-way exchange-based incentive scheme for P2P file sharing systems. By distributing extra tasks to all the peers involved in an n-way exchange, the proposed scheme eliminates prohibitive computation and communication cost on the cooperators, resulting in greater efficiency, effectiveness, and security.

The rest of the paper is organized as follows. In Section 2 we describe related work on incentive mechanisms for P2P systems. In Section 3 we present the existing design of exchange-based incentive scheme by[13], and analyze the complexities of its locating procedure and the limited effectiveness of incentive it provides. In Section 4 we describe the Distributed Exchange Ring Locating(DERL) procedure for our enhanced n-way exchange-based incentive scheme in detail, highlighting several key design issues in terms of efficiency and security. Finally, a conclusion is given in Section 5.

2 Related Work

In monetary schemes, the service consumer pays (real money or virtual currency) to the service provider[14][15]. Despite their flexibility and fine-grained control, monetary schemes are the most expensive to deploy, for they require some underlying accounting and micropayments infrastructure.

Under exchange-based or monetary schemes, peers tend to be short-sighted, for they require an immediate reward in return. If not currently in need, a rich peer has little incentive to cooperate. While reputation-based schemes, featured with delayed reward in return for the serving peer, are naturally immune to this problem.

Under a reputation-based scheme, peers maintain records of the transactional behavior history for others, and base their decision of cooperating or not on these records. Two kinds of reputations are currently in use: private reputation and shared reputation. Private reputation is recorded by individuals, often not shared with others. Private history-based schemes provide ample opportunities for reciprocation between peer-pairs in applications with long session durations and relatively small population; e.g. the Tit-for-tat incentive mechanism in BitTorrent[10]and eMule[11]. Private reputation scheme is cheat-proof, since there is no incentive for a peer to modify its own private records of others. But it does not scale well with the system population. In open file-sharing P2P networks, it is likely that most of time, an individual would be dealing with a total stranger, and their relationship does not last long enough for the server to gain its benefit from the client[16]. Shared reputation addresses these problems through a global reputation propagation procedure, either distributed ([16][18]) or centralized [17][19]. Besides the considerable computation and communication overhead introduced, the sharing of reputation data leaves the hole for collusion[7][20].

3 Exchange-Based Incentive Schemes

Pair-wise exchange incentive schemes based on tit-for-tat strategy have been adopted by real-world P2P file-sharing systems[10][11]. But their performance is limited in systems with large population and great diversity of interest, for it's relatively rare to match users in pairs. In [13], the authors conquer this limit by extending exchanges into an n-way pattern. Incentive for cooperation here is the priority given by the system to exchange over non-exchange transfers. If the peer does not providing anything in return, it will not be included in any exchange transfer.

The n-way scheme's improved effectiveness comes at the expense of the prohibitive ring locating procedure. Each peer maintains a Request Tree (RT) and sends it along with the object request to another peer. A's RT_A consists of itself, as the root, and the set of RTs as the first-level sub-trees, each corresponds to an entry in A's Incoming Request Queue (IRQ). Then, A can initiate an n-way exchange if any peer in the RT_A owns an object desired by A. More specifically, to locate a exchange ring including itself, A has to: 1) before issuing a request for object o_e, inspect the entire RT_A to see if any peer provides o_e; 2) when receiving a request r, inspect its RT_r, for any object that A still wants. If a ring is located, A must also circulate a token along the ring to determine whether everyone is still willing to serve.

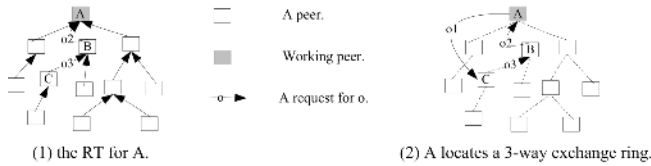

Fig. 1. Ring location in N-way Exchange [13]

For example, in Fig.1, peer A's RT_A is illustrated in (1). If C possesses o_1 wanted by A, then there exists a candidate exchange ring of A, B and C, by which A would get o_1 from C, B gets o_2 from A, and C gets o_3 from A. To locate a potential ring, before A ask others for o_1, first, it will use the resource locating mechanism provided by the system to find out a set of peers, $S(o_1)$, in possession of o_1; then, it inspects its RT_A for any $p \in S(o_1)$, if at this point the RT_B is included in RT_A, it can find C and locate the ring; otherwise if nobody in A's RT_A owns o_1, A will put $(o_1, S(o_1))$ in its wanted list, W_A, and sends a request $Req(A, o_1, RT'_A)$ to others. When the incoming request $Req(B, o_2, RT'_B)$ to A arrives, A will search in its RT_B for every entry in W_A, and finds out that $C \in S(o_1)$, and locates the exchange ring.

While remaining the inherent weak incentive for non-exchange cooperation, this extension from pair-wise to n-way exchange brings forward new issues on efficiency and security. Despite that the authors have empirically determined that it is sufficient to limit the search for cycles to chains of up to 5 predecessors, the cost of communicating the full RT may still be prohibitive for peers with a large number of incoming requests and their neighborhood in the request graph[21]. To make things worse, the above ring locating procedure puts all the burden of searching and checking for exchange rings on the serving peer (i.e. the responding peer). This design can be improved in the following aspects:

1. **Efficiency.** It is expensive, with the worst case overhead of $O(N)$, in which N represents the population of the system. In large open systems, this scale of overhead for a single transaction is unbearable for a single peer. Since for each ring locating operation, a resource locating operation (to find out who owns the one you need)is required. There is additional communication overhead weighing from $O(N)$(flooding) to $O(logN)$ (DHT).
2. **Effectiveness.** It is ineffective as an incentive scheme. And it deduces another kind of free-riding behavior. The more file it shares, the more extra computation and communication tasks a peer has to fulfill. Thus, it is reasonable to believe that it is irrational for the sharing peers to blindly enforce the proposed n-way scheme. Being patient enough, a self-interested peer may prefer to directly forward any incoming request with its RT, in the hope of free-riding on its successor to perform all the task. The disincentive against free-riding if ever provided will be very limited if most cooperative peers behave rationally and refusing to follow the scheme.

3. **Security.** It is insecure because the computation and communication burden on the serving peers can easily be exploited by malicious peers to launch DOS (Denial of Service) attacks against them, or even against the whole system if most behaved peers do follow the scheme.

4 Enhancing the N-Way Scheme with DERL

In this section we present our enhanced exchange-based incentive scheme, based upon the work of[13], for a file sharing system where every request is in the form of a single object (a relatively large and fixed-size block), and peers can download different parts of the same file concurrently. A candidate exchange ring to be used by peers are located during a Distributed Exchange Ring Locating procedure (DERL) by the cooperation of all the peers involved. The fact that no one could depend on others to locate or search for exchange rings strengthens the incentive for the peers to strictly enforce the proposed scheme.

4.1 DERL: The Basic Idea

A pair-wise DERL is straightforward (See Fig. 2) The requestor *Bob* issues a request for object o_2, $req(Bob, o_2)$, to the responder *Alice*. Receiving $req(Bob, o_2)$, *Alice* makes sure that she does own o_2 at the moment, decides the object o_1 she wants currently in exchange for o_2 and then acknowledges *Bob*'s request with $ack(o_1 : Alice|o_2 : Bob)$. If *Bob* owns o_1, he then locates a pair-wise exchange ring $Ring(Alice : o_2|Bob : o_1)$.

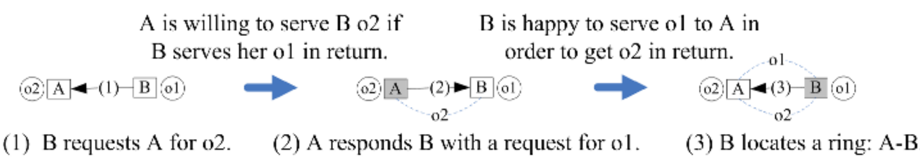

Fig. 2. DERL: A 2-way Example

If *Bob* does not have o_1 (see the first row in Fig. 3), he checks with his IRQ_B, picks up a request $req(Cindy, o_3)$. After checking he has o_3, *Bob* includes a forwarded request for o_1 on behalf of *Alice* in acknowledging *Cindy*, by sending $ack(o_1 : Alice|o_2 : Bob|o_3 : Cindy)$ to her. This message informs *Cindy* that *Bob* agrees to give her o_3, if she delivers o_1 to *Alice*. If *Cindy* owns o_1, she locates a 3-way exchange ring, $Ring(Alice : o_2|Bob : o_3|Cindy : o_1)$. Otherwise, *Cindy* chooses some of its incoming requestors, and forwards them the acknowledge message from *Bob* appended with some extra information about itself. And so the recursive procedure goes on and on until some pre-set limit is reached (e.g. the total number of participants in a single ring preset by the system designer), or a candidate exchange ring is located.

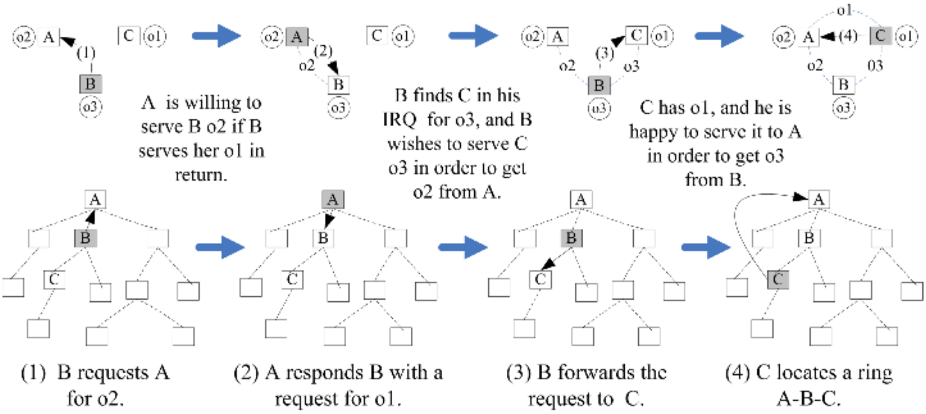

Fig. 3. Understanding DERL with VRTs: A 3-way Example

We argue that DERL covers every valid candidate exchange rings that the Request Tree mechanism proposed in[13]can locate, except those invalid ones whose participants are no longer in need. This fact is better understood if we review DERL in the context of a "virtual RT" (VRT) of the serving peer (see Fig. 3).

4.2 DERL with Object Sets

Several candidate rings rooted at one peer can be located simultaneously by DERL. For example, although in Fig. 3, a 3-way ring has been located already, the DERL procedure on other independent branches of the "virtual request tree" rooted at *Alice* continues, and two more rings $r2 = Ring(Alice : o_2|David : o_3|Emily : o_4|Fred : o_1)$,$r3 = Ring(Alice : o_2|David : o_3|Emily : o_4|Harry : o_1)$ are located as well. These two rings share common edges from *Alice* to *Emily* via *David*, and they are both independent (sharing no common edges) to the previously $r1 = Ring(Alice : o_2|Bob : o_3|Cindy : o_1)$. DERL's ability of locating multiple rings can be used in two ways:

1. Provided with several candidate rings for a single object, *Alice* can choose the best of them based on several carefully chosen metrics, e.g. the trustworthiness of peers involved, the length of the exchange ring, etc.
2. *Alice* can benefit further by issuing a object set $O_A(o_2)$ including all the objects she wants in exchange of o_2. This modification can be fruitful for three reasons.
   (a) It enables *Alice* to evaluate its cost of delivering o_2 to *Bob* when forming $O_A(o_2)$, and maximize the benefit she could get in return by choosing to participate in the exchange ring from which she gets the most desirable one in $O_A(o_2)$.

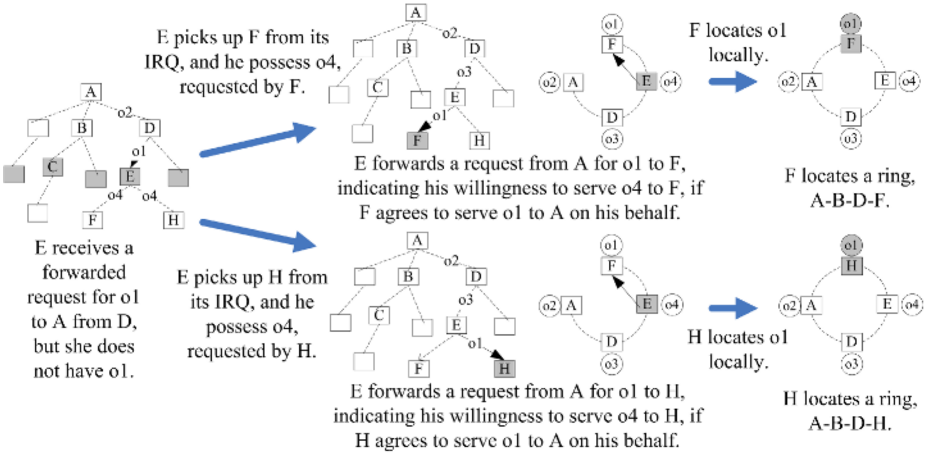

Fig. 4. Multiple Rings Located in a Single Instance of DERL: An Example

(b) *Bob* now has much greater chance of locating an ring with *Alice* since he would succeeds if he owns any $o_x \in O_A(o_2)$, and so does all his successors (the peers located under *Bob* in its VRT). If *Bob* owns several objects in $O_A(o_2)$, he can bargain with her by delivering the one with the least cost.

(c) Different exchange rings can be located simultaneously for different objects in $O_A(o_2)$, further benefiting *Alice*.

The pseudo-code description of DERL with object sets is given as follows:(For any peer p_i in the system, we use p_{ic} and p_{is} to denote its dual identities participating DERL, as the requestor and the server, respectively.)

1. **As the requestor in a transaction:** (p_{ic})
   (a) p_{ic} locates the desired object o_{ic} to its current provider p_s, sends $req(p_{ic}, o_{ic})$ to p_s, and waits for the response.
   (b) Upon receiving $ack(O_{p_1}(o_2) : p_1|o_2 : p_2|\cdots|o_{n-2} : p_{n-2}|o_s : p_s|o_{ic} : p_{ic})$ from p_s, p_{ic} searches locally for any $o_{ix} \in O_{p_1}(o_2)$.
      i. If p_{ic} finds a set of objects, it locates a set of candidate n-way exchange ring in the form of $Ring(p_1 : o_2|p_2 : o_3|\cdots|p_s : o_{ic}|p_{ic} : o_{1x})$. p_{ic} exits.
      ii. Otherwise, p_{ic} searches its IRQ for any request from p_1. If he finds a local request set $S_{o_i}(p_1)$, P_{ic} locates a set of candidate n-way exchange ring in the form of $Ring(p_1 : o_2|p_2 : o_3|\cdots|p_s : o_{ic}|p_{ic} : o_{1y})$ for each $o_{1y} \in S_{oi}(p_1)$. p_{ic} exits.
      iii. Otherwise, p_{ic} selects a subset of requestors from its own IRQ, $S_{p_i}(O_{p_1}(o_2), p_1)$. To each $p_{ix} \in S_{p_i}(O_{p_1}(o_2), p_1)$requesting for o_x, if p_{ic} currently owns o_x, p_{ic} sends $ack(O_{p_1}(o_2) : p_1|o_2 : p_2|\cdots|o_{n-2} : p_{n-2}|o_s : p_s|o_{ic} : p_{ic}|o_x : p_{ix})$. p_{ic} exits.

2. **As the responder in a transaction:** (p_{is}) Receiving $req(p_x, o_x)$ from p_x, p_{is} checks locally for o_x. If it succeeds and is willing to exchange o_x with $p_x$for any $o_{sx} \in O_{p_{is}}(o_x)$, p_{is} acknowledges p_x with $ack(O_{p_{is}}(o_x) : p_{is}|o_x : p_x)$.

4.3 Selecting Compatible Rings After DERL

Although three exchange rings have been located during the DERL procedure in Fig. 4, only one of them will be put into real use, since normally *Alice* wouldn't want to download o_1 more than once. In other words, these three are mutually incompatible. We can define the compatibility relationship on $\mathbb{R}$ as follows:

Definition 1. *Upon the set of candidate exchange rings* $\mathbb{R}$*, the set of peers in the system* $\mathbb{P}$*, and the total set of exchangeable objects in the system* $\mathbb{O}$*, We define the function* $o : \mathbb{R} \times \mathbb{P} \mapsto \mathbb{O} \cup \{null\}$ *as follows:*

$$\forall r \in \mathbb{R}, \forall p \in \mathbb{P} : \qquad o(r,p) = \begin{cases} o_x & , \textit{if } p \textit{ participates } r \textit{ to get } o_x \\ null & , \textit{otherwise} \end{cases} .$$

Definition 2. *The compatibility relationship between any two candidate rings:*

$$\forall r_i, r_j \in \mathbb{R}, ri \neq r_j :$$
$$Com(r_i, rj) = False \quad iff \quad \exists p \in \mathbb{P} : o(r_i, p) = o(r_j, p) \neq null.$$

For example, the three rings illustrated in Fig. 3 and Fig. 4, r_1, r_2 and r_3 are mutually incompatible, since they all provides o_1 to *Alice* if put to use.

Note that we assume a rational peer will never download an object twice. Therefore, at any point of time, any two rings currently in use in the system must be compatible. The compatibility relationship can be used by participants when making decision in the exchange ring validating procedure following DERL with candidate exchange rings. e.g. if *Alice* has chosen r_1, then she would refuse to participate r_2 or r_3.

4.4 Comparison and Analysis

First of all, DERL is a distributed procedure, involving all the peers along the candidate ring. By distributing the burdensome locating task to the requestor and its requestors recursively, DERL overcomes the prohibitive overhead on a single peer of the original locating procedure in[13], without losing the capacity to locate every possible candidate ring. DERL further improves its efficiency by eliminating outdated braches (request made earlier but has become invalid by the time the locating procedure is executed) in the VRT automatically since a rational peer never submit a request to another peer for what he has already obtained from a third peer.

Second, in [13]'s design, to locate an exchange ring, the requestor's entire *RT* tree (or a large part of it) is transferred along with every outgoing request. This incurs a huge communication overhead for both parties involved. In our design, this information is reduced to minimum as each peer only maintains its own Incoming Request Queue (IRQ) locally. However, during the DERL, the responder's request message as forwarded by the requestors recursively is growing continuously until the preset ring length limit is reached. The average communication overhead has been reduced from $O(2^{\mathcal{L}})$ to $O(\mathcal{L})$, where $\mathcal{L}$ denotes the average or preset ring length.

Third, DERL provides much stronger incentive for resource-sharing since the requestor can no longer free-ride on the responder forring locating. On one hand, it becomes more convenient for responders to share, encouraging more peers to cooperate or the cooperating peers to cooperate more. On the other hand, since DERL requires a consistent interaction for the requestors with others, it becomes more energy-consuming to submit a request and participate into a exchange transfer, discouraging lavish misbehavior.

Finally, DERL is more robust and efficient, for several rings rooted at one peer can be located simultaneously during a single instance of DERL's execution. While in the previous work, due to the prohibitive cost for full RT tree searching, root peer exits searching for an given object as soon as one candidate exchange ring is located. In a dynamic environment like P2P networks, where peers come and go frequently, DERL's capacity to locate more candidates may be the key to succeed. And more candidates indicates more flexibility for the requestor in choosing the provider for a given object, which allows room for more sophisticated security enhancement. For example, integrated with a proper-designed reputation system, DERL can also be used to mitigate the impact of fake object attacks, where dishonest peers try to spoof others into exchanging with fake objects.

5 Conclusion

We presented our enhanced version of n-way exchange-based incentive scheme for P2P file-sharing systems in this paper. Using the Distributed Exchange Ring Locating procedure (DERL), the scheme distributes extra tasks to all the peers in an exchange, eliminates the prohibitive cost on the cooperators, and results in greater efficiency and effectiveness in terms of providing strong incentive to rational peers both for cooperating and enforcing the proposed scheme.

References

1. M. Feldman and J. Chuang. Overcoming Free-Riding Behavior in Peer-to-Peer Systems. In ACM SIGecon Exchanges, Vol. 5, No.4, 2005, pp. 41-50.
2. E. Adar and B. Huberman. Free Riding on Gnutella. FirstMonday, 2000.
3. D. Hughes, G. Coulson, and J. Walkerdine. Freeriding. on Gnutella Revisited: the Bell Tolls? IEEE. Distributed Systems Online, 6(6), 2005.
4. S. Saroiu, P. Krishna Gummadi, and S. D. Gribble. A Measurement Study of Peer-to-Peer File Sharing Systems. In Proc. of the 9th Annual Multimedia Computing and Networking (MMCN), 2002.

5. M. Yang, Z. Zhang, X. Li, et al. An Empirical Study of Free-Riding Behavior in the Maze P2P File-Sharing System. In Proc. of the 4th Int'l. Workshop on Peer-To-Peer Systems (IPTPS), 2005.
6. J. Pouwelse, P. Garbacki, D. Epema, et al. The BitTorrent P2P File-sharing System: Measurements and Analysis. In Proc. of the 4th Int'l. Workshop on Peer-To-Peer Systems (IPTPS), 2005.
7. M. Feldman, K. Lai, I. Stoica, et al., Robust Incentive Techniques for Peer-to-Peer Networks. In Proc. of the 6th ACM conference on Electronic Commerce, 2004.
8. P. Obreiter, B. Konig-Ries, M. Klein. Stimulating Cooperative Behavior of Autonomous Devices - An Analysis of Requirements and Existing Approaches. In Proc. of the 2nd Int'l. Workshop on Wireless Information Systems (WIS), 2003, pp.71-82.
9. P. Obreiter, J. Nimis. A Taxonomy of Incentive Patterns-The Design Space of Incentives for Cooperation. In Proc. of the 2nd Int'l. Workshop on Agents and P2P Computing, Melbourne, Australia, Springer-Verlag, LNCS 2872, 2003, pp.678-685.
10. B. Cohen. Incentives Build Robustness in BitTorrent. In Proc. of the 1st Workshop on Economics of Peer-to-Peer Systems (P2PECON), 2003.
11. The EMule Project. http://www.emule-project.net
12. S. Jun and M. Ahamad. Incentives in BitTorrent Induce Free Riding. In Proc. of the ACM Symposium on Communications Architectures and Protocols (SIGCOMM), 2005.
13. Kostas G. Anagnostakis and Michael B. Greenwald, Exchange-based Incentive Mechanisms for Peer-to-Peer File Sharing. In Proc. of the 24th int'l. Conference on Distributed Computing Systems (ICDCS), 2004, pp. 524-533.
14. P. Golle and K. Leyton-Brown. Incentives for Sharing in Peer-to-Peer Networks. In Proc. of the 3rd ACM Conference on Electronic Commerce (EC), 2001.
15. Ioannidis, J., Ioannidis, S., Keromytis, A., et al. Fileteller: Paying and Getting Paid for File Storage. In Proc. of the 6th Int'l. Conference on Financial Cryptography (FC),2002.
16. Ernesto Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, Pierangela Samarati: Managing and Sharing Servents' Reputations in P2P Systems. IEEE Trans. Knowl. Data Eng. 15(4): 840-854,2003.
17. M. Gupta, P. Judge, and MH Ammar. A Reputation System for Peer-to-Peer Networks, in Proc. of the 13th Int'l. Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV), 2003.
18. Kamvar, Schlosser, Garcia-Molina. The EigenTrust Algorithm for Reputation Management in P2P Networks. In: Proc. of the 12th. Int'l Conference on World Wide Web (WWW), 2003.
19. M. Yang, H. Chen, BY Zhao, et al. Deployment of a Large-scale Peer-to-Peer Social Network. In Proc. of WORLDS, San Francisco, CA, 2004.
20. Q. Lian, Z. Zhang, M. Yang, et al. An Empirical Study of Collusion Behavior in the Maze P2P File-Sharing System. Microsoft Research Technical Report, MSR-TR-2006-14, 2006.
21. Kostas G. Anagnostakis. Exchange Mechanism and Cooperative Computing Systems Design. Doctoral Dissertation, University of Pennsylvania, 2005.