A Cooperative Peer Clustering Scheme for Unstructured Peer-to-Peer Systems

Satoshi Fujita[1]

[1]Department of Information Engineering, Hiroshima University, Japan
fujita@se.hiroshima-u.ac.jp

ABSTRACT

This paper proposes a peer clustering scheme for unstructured Peer-to-Peer (P2P) systems. The proposed scheme consists of an identification of critical links, local reconfiguration of incident links, and a retaliation rule. The simulation result indicates that the proposed scheme improves the performance of previous schemes and that a peer taking a cooperative action will receive a higher profit than selfish peers.

KEYWORDS

Unstructured P2P, Peer Clustering, Local Reconfiguration, Retaliation Rule

1. INTRODUCTION

Peer clustering is a key operation for fully distributed systems such as wireless ad hoc networks and unstructured Peer-to-Peer (P2P) systems [4,7,8,9,10,11]. The objective of peer clustering is to reconfigure the structure of an overlay network in such a way that the specific peers becomes closer without increasing the total number of links in the network.

The performance of peer clustering schemes in unstructured P2Ps is generally measured by the hit rate of a search task and/or the cost required for specific tasks such as message routing, streaming, and others. Although there are several peer clustering schemes proposed in the literature [1,5,6], the performance of those schemes is severely affected by the “criticalness” of links in the overlay. For example, in unstructured P2Ps, a file search is realized by flooding a query message through an overlay by setting an appropriate TTL (Time-to-Live) to each query. Hence, the removal of a critical link would cause an unreachability of queries to their destination, which significantly degrades the hit rate of the overall search process. On the other hand, many of existing clustering schemes could not tolerate a situation in which a peer which fully utilizes its incident links refuses an additional request for a connection even if it has a neighbor to have enough capacity. Such observations motivate us to develop a peer clustering scheme in which participating peers wish to cooperate with each other, in such a way that the profit of all peers are kept sufficiently high, important links will be given a high priority, and a peer with high capacity could support the connection of other low-capacity peers.

In this paper, we propose a peer clustering scheme to attain such goals. More concretely, after reviewing related work in Section 2, we will propose a cooperative peer clustering scheme for unstructured P2Ps. The basic idea of the scheme is to use the notion of retaliation similar to Tit-for-Tat strategy which has been widely used in many P2P systems including BitTorrent. The performance of the proposed scheme is evaluated by simulation. The result of simulations

DOI : 10.5121/ijp2p.2013.4201

indicates that the proposed scheme certainly improves the performance of previous schemes with respect to the hit rate and a peer taking a cooperative action will receive a higher profit than selfish peers.

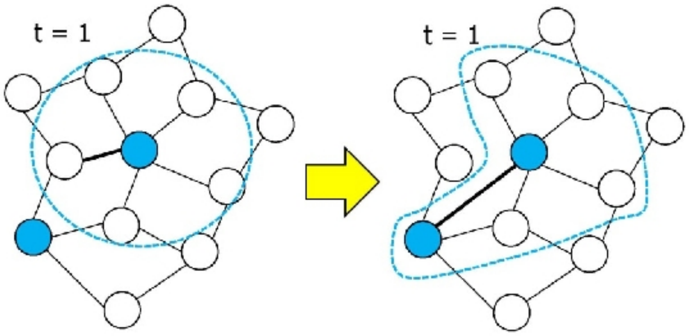


Figure 1.Reconfiguration of overlay network.

2. RELATED WORK

Cholvi et al. [1]proposed a clustering scheme in which two peers are connected by a link when they mutually recognize their counterpart as an acquaintance, where peer “a” recognizes peer “b” as an acquaintance if “b” provides “a” a file requested by “a.”Raftpoulou and Petrakis proposed a scheme based on the similarity of interest [5], where interest of users is defined as the type of files, and is represented by a characteristic vector in an appropriate vector space. Sripanidkulchai et al. proposed a scheme based on the notion of shortcuts which are temporally established between peers while conducting a file search [6]. The reader should note that in all of the above three schemes, each peer conducts a (re)establishment of links in a selfish manner and does not consider the benefit of other peers while conducting such a (re)establishment.

Tit-for-tat (TFT) is a common strategy used in two-players games, which is informally described as follows: *Unless provoked, the player will always cooperate, and if provoked, the player will retaliate*. It is widely recognized that under such an equivalent retaliation strategy, a selfish player could not obtain enough profit compared with a cooperative player who tries to keep the profit of the other players while trying to increase its own profit. TFT strategy has already been used in many P2P applications. For example, in BitTorrent [2], each shared file is divided into small fragments called pieces, and is downloaded from the network by repeating an exchange of pieces among nearby peers, where TFT is used in such a way that a peer who uploaded a piece to other peers is granted a right to download necessary pieces from other peers. As another example, in Garbacki’s protocol [3], a peer is granted to use the communication bandwidth of other peers if it contributes to those peers by providing its communication bandwidth.

3. PROPOSED METHOD

3.1. Critical Links

In this paper, for simplicity, we assume that each peer belongs to exactly one community sharing the same interest[1]. As for the definition of interest and community, we adopt a simple model in

[1] We also assume that such a community is constructed merely implicitly and it is not possible to register all such communities to a centralized computer such as tracker and index server used in many existing P2P systems.

order to concentrate on the effect of the retaliation in peer clustering (see Section 4.1 for the detail of simulation model). In addition, we assume that each peer wants to collect as many peers belonging to the same community within a predetermined “visible” region as possible, and regard the number of such visible peers as the profit received through a peer clustering. More concretely, peers in each community are initially distributed over an overlay network in an arbitrary manner, and during a clustering, they try to reconfigure the network in such a way that the number of visible peers is maximized, without increasing the total number of links and without reducing the number of visible peers for the other peers. See Figure 1 for illustration. The left figure shows the initial overlay in which two blue peers are not visible with each other with TTL one and the right figure shows the overlay after conducting a reconfiguration so that two blue peers are visible with each other.

Let t be an integer representing the limit for such a visible region, i.e., t corresponds to the TTL of queries issued by each peer. Let A be a community. As a formal definition of the criticalness of links, the notion of t-criticalness is now defined as follows:

Definition1 LetubeapeerincommunityAandebealinkinG. eis said to be t-critical for u if there is a peer v in community A such that: 1) the distance between u and v in G = (V,E) is at most t, and 2) the distance between u and v in G’= (V,E −{e}) is at least t+1.

In what follows, a t-critical link for some peer is simply referred to as t-critical, and we often omit parameter t if it is clear from the context.

3.2. Recognition of Critical Links

In the proposed scheme, query and query response play an important role to recognize critical links. Before issuing a query, the originator of the query attaches its interest to the query. It then broadcasts the query to all peers within a fixed TTL, where each copy of the query records: 1) the length of a shortest path from the originator (i.e., hop count) and 2) ID of peers existing on the forwarding path. Suppose that peer u receives a query from an adjacent peer. If it satisfies one of the following two conditions, u returns a query response to the originator, and otherwise, it simply forwards a copy of the received query to its neighbors as long as it did not exhaust the TTL:

- If it holds a file matching the given query, or
- If it has a similar interest to the originator of the query.

Query response is returned to the originator through the forwarding path in a reverse direction. By analyzing query responses received from adjacent peers, the originator can identify a peer which has a similar interest to the originator and is located at distance t from him.

After identifying such critical links in the network, each peer notifies it to all peers in its range of TTL by attaching it to the queries issued in the succeeding steps. By this notification, each peer can recognize the criticalness of links incident to the peer.

3.3. Recognition Rule

In the proposed clustering scheme, the notion of **mate** plays an important role. Two peers “a” and “b” sharing the same interest are said to be mate if “a” is incident on a critical link for “b,” and “b” is incident on a critical link for “a.” The proposed reconfiguration rule is designed in such a way that each peer tries to keep critical links for its mates. More concretely, each peer can remove its incident links according to the following rule:

1. A link which is not critical for any mate can always be removed.
2. If all incident links are critical for some mate, then with probability 1/k for some integer k (≥ 1), it can remove one of such links.

In the evaluation shown in the next section, we will fix parameter k to 20 according to the result of preliminary experiments.

3.4. Retaliation

In order to realize an effective retaliation to a treachery, in the proposed scheme, we use a tracker to keep the history of reconfigurations conducted by the participant peers[2]. Concrete procedure is described as follows.

Step 1: Suppose that peer “a” removes an incident link connecting to peer“b.” After completing such a removal, peer “a” notifies the fact of removal to the tracker with the following information: 1) address of “a,” 2) address of “b,” and 3) interests of “b.” In the following, we call it an update information. Received update information is stored at the tracker for a predetermined time period.

Step 2: Each peer “c” periodically requests the tracker to send back a list of update information. After receiving it, “c” identifies a set of peers which removed a critical link for “c” (the set may be empty if all peers are cooperative). Let “d” be a peer contained in the identified set. If the number of critical links for “c” which are removed by peer “d” exceeds a predetermined threshold, “c” reports the fact to the tracker. If the number of reports concerned with peer “d” exceeds another threshold, the tracker appends “d” to the black list.

Step 3: Each peer periodically requests the tracker to send back the black list. Then, for each peer “d” contained in the list, the link connecting to “d” is forced to be removed (if any), and any request received from “d” will be refused.

4. EVALUATION

4.1. Setup

Fix a set of 20 communities, and associate each peer with a random subset of four communities. We say that two peers are friends if their corresponding subsets have a non-empty intersection (by definition, a mate is a friend, but the reverse is not true). The number of peers is fixed to 1000. Each peer has a file associated with each community in the subset, and each query issued by the peers designates a community concerned with the requested file.

The performance of a clustering scheme is evaluated by using the following two metrics:

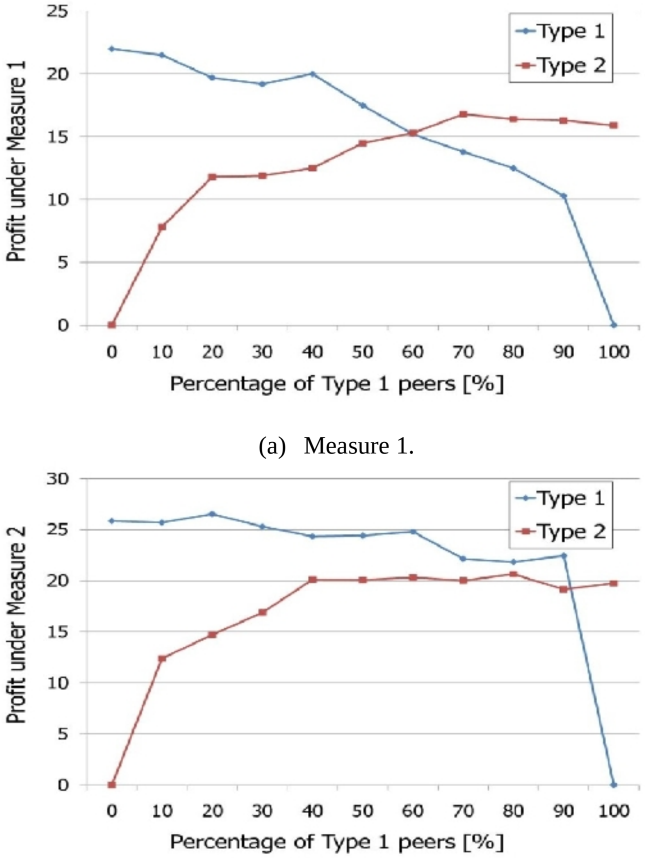
(1) The number of friends within t hops from the examined peer (Measure 1), and
(2) The number of friends weighted by an inverse of the distance from the examined peer (Measure 2),

[2] Note that it is reasonable to assume the existence of such a centralized computer, since many P2P systems such as BitTorrent rely on the tracker to realize a join of new peers to the network.

where by letting N(h) be the number of friends at distance h, the latter metric is formally described as

$$\sum_{h=1}^{\infty} \frac{N(h)}{h}$$

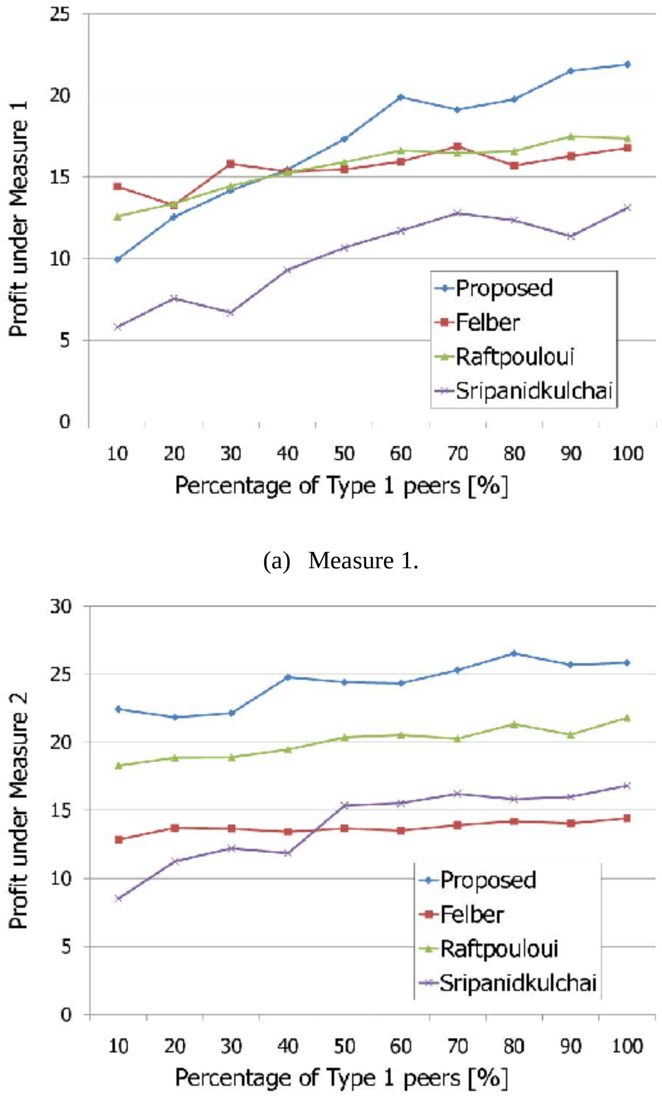
In the following, we assume that each peer follows a predetermined clustering scheme in establishing a link. More concretely, in Section 4.2, we will use a simple clustering scheme which tries to establish a link to a friend discovered during a flooding of queries, and in Section 4.3, we examine several clustering schemes proposed in the literature. On the other hand, as for the removal of links, we will distinguish two cases, i.e., whether it follows the proposed disconnection rule or not. A peer which follows the rule is called Type 1, and a peer which does not follow the rule is called Type 2. In the simulation, we assume that x % of peers are of Type 1 and the remaining peers are of Type 2, where parameter x varies from 10 to 100. The retaliation rule is uniformly applied to both types of peers. Thus, it is expected that although peer of Type 2 could receive a high profit within a short time period, as the elapsed time increases, the profit of Type 1 peers becomes higher than the profit of Type 2 peers. In the simulation, we fix the simulation time to 5 min. During this time period, each peer repeats the issue of a query and a reconfiguration of incident links about 150 times.



(a) Measure 1.

(b) Measure 2.

Figure 2.Performance of the proposed scheme.



(a) Measure 1.

(b) Measure 2.

Figure 3.Comparison with previous schemes.

4.2. Effect of Proposed Disconnection Rule

At first, we evaluate the effect of the proposed cooperative disconnection rule. Figure 2 summarizes the result for k = 20 and t = 3, where the horizontal axis is the percentage x of Type 1 peers. As shown in the figure, in both metrics, the profit of Type 1 decreases as x decreases, while that of Type 2 increases as x decreases. Two curves cross around x = 40% in Measure 1 and 10% in Measure 2, where in general, Measure 2 evaluates the schemes more accurately than Measure 1, since it reflects the distribution of friends beyond TTL.

Thus, although a detailed game theoretic analysis is left as a future work, if there are more Type 1 peers than the crossing point, a reasonable peer should take a cooperative action to increase its own profit since it provides the peer a higher profit. In addition, peers at the crossing point should take a cooperative action since it increases the chance of obtaining a higher profit, where the badness of Type 2 peers is apparently due to the retaliation process and selfish behavior conducted by other Type 2 peers.

4.3. Effect of Cooperative Disconnection in Other Schemes

Next, we evaluate the impact of the proposed disconnection rule in existing clustering schemes described in Section 2. Figure 3 summarizes the result for k = 20 and t = 3, where the horizontal axis is the percentage of Type 1 peers and the vertical axis is the profit averaged over all peers including Type 1 and Type 2.

In Cholvi’s scheme, two peers are connected by a link if they mutually recognize their counterpart as an acquaintance. In other words, the criteria for establishing a connection is much higher than the random scheme examined in the last subsection although the possibility of removing a link by an incident peer is rather small. As a result, although it beats the randomized scheme with respect to Measure 1 for small x’s, the profit does not glow as rapidly as the randomized scheme for larger x’s. In addition, as for Measure 2, the profit of the randomized scheme is almost twice of the Cholvi’s scheme.

The heuristic adopted in the Raftpolou’s scheme conflicts with the cooperative behavior of Type 1 peers. In Raftpoulou’s scheme, each peer acquires the information of remote peers through long-range links, i.e., it uses those links in keeping the scope of the participant peers, while it reconfigures the overlay based on the similarity of their interest. Thus the effect of reconfigurations becomes small if many peers act cooperatively. In fact, as shown in the figure, the randomized scheme outperforms the Raftpolou’s scheme for large x’s; e.g., the amount of improvement is 12% for Measure 1 and 21% for Measure 2.

The superiority of the randomized scheme can also be observed in a comparison with the Sripanidkulchai’s scheme; e.g., the amount of improvement is 69% for Measure 1 and 74% for Measure 2. The key idea of the Sripanidkulchai’s scheme is to use shortcuts in realizing effective reconfigurations, where shortcut is a tentative link established during a file exploration and will be removed after completing the exploration. Thus, even if it would be t-critical for some peer, a shortcut is easily removed in many cases, and such a selfish behavior increases the frequency of invocations of the retaliations, which degrades the performance of the overall scheme.

5. CONCLUDING REMARKS

This paper proposed a cooperative peer clustering scheme for unstructured P2Ps based on the notion of retaliation. The performance of the proposed scheme is evaluated by simulation, and the result of simulations indicates that it certainly improves the performance of conventional schemes particularly when the percentage of cooperative peers is large. A future work is to provide a theoretical analysis of the proposed scheme, including the analysis of the convergence speed.

ACKNOWLEDGEMENTS

The author would like to thank Mr. Aoki for his contribution to conduct simulations.

REFERENCES

[1] V. Cholvi, P. Felberand E. Biersack, (2004) “Efficient Search in Unstructured Peer-to-Peer Networks,”Proc. 16th Annual ACM Symp.on Parallelism in Algorithms and Architectures, pp.271-272.

[2] B. Cohen, (2003) “Incentives Build Robustness in BitTorrent,”bittorrent.org.

[3] P.Garbacki,D.H.J.EpemaandM.V.Steen, (2007) “AnAmortizedTit-For-Tat Protocol for Exchanging Bandwidth instead of Content in P2P Networks,” Proc. 1st International Conference on Self-Adaptive and Self-OrganizingSystems, pp.119-128.

[4] Available at http://en.wikipedia.org/wiki/Gnutella.

[5] P. Raftopoulou and E. G. M. Petrakis, (2008) “iCluster: A Self-organizing OverlayNetwork for P2P Information Retrieval,”Advances in Information Retrieval:Springer, pp.65-76.

[6] K.Sripanidkulchai,B.MaggsandH.Zhang, (2003) “Efficientcontentlocationusing interest-based locality in peer-to-peer systems,”Proc. INFOCOM, pp.2166-2176.

[7] M. Xu, G.-Z. Liu, (2011) “Building self-adaptive Peer-to-Peer overlay networks with dynamic cluster structure,” Proc. 13th International Conference on Communication Technology (ICCT), IEEE, pp.520-525.

[8] D. B.Khedher, (2012) “A Peer-to-Peer Self-Organizing Scheme for Multiparty Session,”Proc. International Conference on Communication (ICC 2012), IEEE, pp. 6535-6539.

[9] H.-C.Jang, L.-J.Tzeng, (2012) “Affinity Propagation with File Similarity based Clustering for P2P File Sharing in VANET,”Proc. the 15th International Symposium on Wireless Personal Multimedia Communications (WPMC 2012), pp.70-74.

[10] Z.-J. Deng, W. Song, X.-F.Zheng, (2010) “P2PKMM: A Hybrid Clustering Algorithm over P2P Network,”Proc. the3rd International Symposium on Intelligent Information Technology and Security Informatics (IITSI 2010), pp. 450-454.

[11] S. Aslam, I. Kazmi, M. Y. Javed, M.S. Anwar, (2010) “Cluster based peers configuration with multiple physical parameters using HCNP in Peer-to-Peer overlay networks,” Proc.2010 International Conference on Computer Applications and Industrial Electronics (ICCAIE), pp.142-147.

AUTHOR

Dr. Satoshi Fujita received the B.E. degree in electrical engineering, M.E. degree in systems engineering, and Dr.E. degree in information engineering from Hiroshima University in 1985, 1987, and 1990, respectively. Currently, he is a Professor at the Institute of Engineering, Hiroshima University. His research interests include communication algorithms in interconnection networks, parallel algorithms, graph algorithms, and parallel and distributed computer systems. He is a member of the IEICE, IPSJ, SIAM Japan, IEEE Computer Society, and SIAM.