Proposal and Evaluation of a Cooperative Mechanism for Pure P2P File Sharing Networks

Junjiro Konishi, Naoki Wakamiya, and Masayuki Murata

Graduate School of Information Science and Technology, Osaka University
1–5 Yamadaoka, Suita-shi, Osaka 565–0871, Japan
{j-konisi, wakamiya, murata}@ist.osaka-u.ac.jp

Abstract. To provide application-oriented network services, a variety of overlay networks are deployed over physical IP networks. Since they share and compete for the same physical network resources, their selfish behaviors affect each other and, as a result, their performance deteriorates. Our research group considers a model of overlay network symbiosis, where overlay networks coexist and cooperate to improve their application-level quality of service (QoS) while sustaining influences from the physical network and other overlay networks. In this paper, we propose a mechanism for pure P2P networks of file-sharing applications to cooperate with each other. In our proposal, cooperative peers establish logical links among two or more P2P networks, and messages and files are exchanged among cooperative P2P networks through these logical links. For efficient and effective cooperation, we also propose an algorithm for selection of these cooperative peers. Simulation results show that our proposed mechanism improves the search efficiency of P2P file-sharing applications and reduces the load in P2P networks.

1 Introduction

To provide application-oriented network services, a variety of overlay networks are deployed over physical IP networks. Each overlay network independently measures network conditions such as the available bandwidth and latency through active or passive measurement schemes. Based on its observations, each overlay network controls traffic, chooses routes, and changes topologies in a selfish manner to satisfy its own application-level QoS. Since overlay networks share and compete for the same physical network resources, their selfish behaviors affect each other and their performance deteriorates [1, 2]. For example, to communicate faster with other nodes, a node measures bandwidth and latency to other nodes and changes its neighborship accordingly. As a result, the load in the physical network dynamically changes and consequently the quality of communication perceived by other overlay networks which compete for the same links and routers in the physical networks deteriorates. Those affected overlay networks then adapt data rate, routes, and topologies to satisfy or improve their application-level QoS. This further affects other overlay networks and it causes frequent changes of routing and introduces congestions in the physical network. Finally, the selfish behavior of overlay networks trying to improve their application-level QoS in fact results in the deterioration of application-level QoS.

Recently there are several publications on cooperative overlay networks to enhance their collective performance and efficiently utilize network resources [3–6]. In [3], the authors investigate a spectrum of cooperation among competing overlay networks. For example, they propose an architecture where overlay networks cooperate with each other in inter-overlay routing where a message from one overlay network is forwarded to another which provides a shorter path to the destination. In [4], mechanisms are proposed to exchange information among overlay networks without knowing the destination addresses by using an overlay network called i3 (Internet Indirection Infrastructure) network. The i3 network is an network architecture consisted of some servers. In the i3 network, a user sends *trigger* messages with a service identifier and user's address to the i3 network. A service provider sends *packet* messages with a service identifier to the i3 network. The i3 network transfers *packet* messages to users whose *trigger* messages have the same or similar service identifier.

The analysis on coexistence of competitors has been investigated in the field of biology. In an ecosystem, organisms live together in the same environment with direct and/or indirect interactions with each other. In [7], the authors established a mathematical model of the metabolic pathways of bacterial strains to elucidate mechanisms of coexistence of living organisms of closely related species. They revealed that the coexistence emerged not only from interactions among competitors, but also from changes of their internal states.

Taking inspirations from biology, our research group considers the symbiosis among competing overlay networks [8]. We regard an overlay network as an organism. In the model of symbiotic overlay networks, overlay networks in a system evolve, interact with each other, and dynamically change internal structures, but they still behave in a selfish manner, as living organisms in the same environment do. Overlay networks meet and communicate with each other in a probabilistic way. Overlay networks that benefit from each other reinforce their relationship, eventually having many inter-overlay links, and merging one overlay network. Otherwise, they separate from each other. All evolutions, interactions, and internal changes are performed in a self-organizing way. Each node independently decides its behavior based only on locally available information. Symbiosis among overlay networks emerges as a consequence of the independent and autonomous behaviors of nodes and networks.

For this purpose, we need mechanisms for overlay networks to communicate with each other as in biological systems. In this paper, we propose a mechanism for pure P2P networks of file-sharing applications to interact and cooperate with each other in an efficient and effective way. In a P2P network, hosts called peers directly communicate with each other and exchange information without the mediation of servers. According to user's intention, peers consisting in a P2P network behave on its own decision as an individual does in a group or society. One typical example of P2P applications is a file-sharing system. Gnutella and

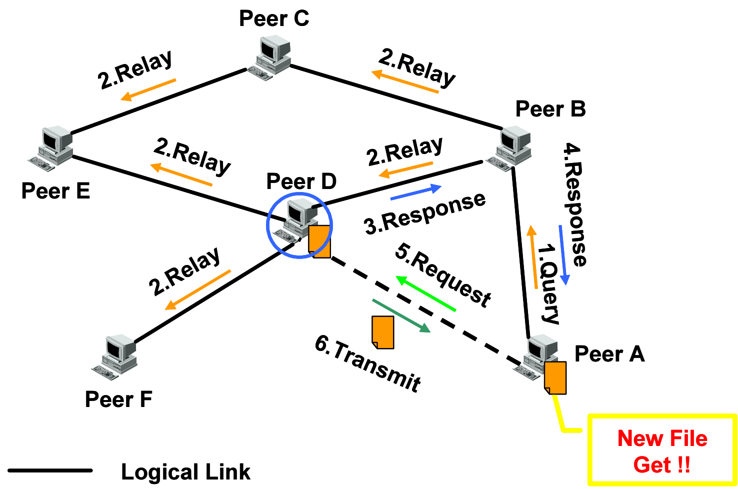

Fig. 1. Flooding in a pure P2P file-sharing network

Winny are categorized as pure P2P networks without a server for searching files. Thus, a peer has to find the desired file by itself by emitting a query message into the network. Other peers in the network reply to the query with a response message and relay the query to their neighbor peers (Fig.1). Flooding, in which a peer relays a query message to every neighbor peer, is a powerful scheme for finding a desired file in a P2P network. However, it has been pointed out that the flooding scheme lacks scalability because the number of query messages that traverses a network significantly increases with the growth in the number of peers [9].

The cooperation among pure P2P networks is accomplished by exchanges of search and reply messages among them through logical connections established among so-called cooperative peers. With such cooperation, we can expect that search messages are disseminated more effectively and a peer finds file more efficiently. Since a peer receives more reply messages for a file, it can choose a more appropriate peer, i.e., faster and more reliable, among many candidate peers, leading to a higher application-level QoS. Furthermore, when a P2P network is disconnected by failures or disappearance of peers, search and reply messages can propagate among separated parts of the P2P networks through cooperative P2P networks. However, to accomplish the efficient and effective cooperation without introducing much load on logical and physical networks, some careful considerations must be made. For example, if a cooperative peer is located at the edge of a P2P network, it has to set a large TTL (Time to Live) value for search messages to spread over the network. As a result, the number of rejected duplicated search messages over P2P networks increases. They waste network bandwidth and causes network congestions. Therefore, we propose an algorithm to choose appropriate cooperative peers. We should note here that a cooperative mechanism should leave peers selfish. Cooperation should emerge from selfish behavior of peers who want to enhance and improve their own QoS. We give some considerations on incentives that a peer begins cooperation.

The rest of this paper is organized as follows. In Section 2, we propose a mechanism for cooperation among pure P2P networks of file-sharing applications. In Section 3, we evaluate our mechanism through several simulation experiments from the viewpoint of the reachability of search messages, the number of found files, and the load on peers. Finally, we conclude the paper and describe future works in Section 4.

2 Cooperative Mechanism for Pure P2P File-Sharing Networks

In this section, we propose a mechanism for pure P2P networks of file-sharing applications to cooperate with each other in an efficient and effective way. In the cooperation of pure P2P networks of file-sharing applications, a logical link is first established between designated peers, called cooperative peers, which are selected among candidate peers in each P2P network. Candidate peers are those which are willing to play the role for cooperation to enhance and improve their own QoS. And then search and reply messages are transmitted through the logical link between cooperative peers (Fig.2).

The mechanism consists of the following steps. First, a peer in a P2P network is promoted to a candidate peer by running a cooperative program. Second, candidate peers construct a candidate network to exchange information for the selection of cooperative peers. Third, a tentative cooperative peer is selected in candidate peers, and then it confirms whether it is appropriate as a cooperative peer or not. Finally, after the confirmation, a tentative cooperative peer is promoted to a cooperative peer. We describe in the following the selection of cooperative peers, the discovery of other P2P networks, the decision of starting a cooperation, the relay of messages and the transfer of files, and the decision of finishing a cooperation in detail.

2.1 Establishing a Candidate Network

When a peer is not satisfied with an application-level QoS received from a P2P network of file-sharing application, it considers to enhance and improve its application-level QoS by its own decision. For example, when a peer cannot find a desired file at all, when a peer cannot find enough number of files against its query, or when a peer cannot tolerate the delay in retrieving a file from a provider peer, a peer, i.e., a user should have some frustrations. A peer will consider that it can receive the higher QoS by connecting to another P2P network which provides it with the higher probability of successful search, the larger number of provider peers, and the smaller delay in file retrieval. In such a case, intending to enhance and improve its application-level QoS, a peer runs

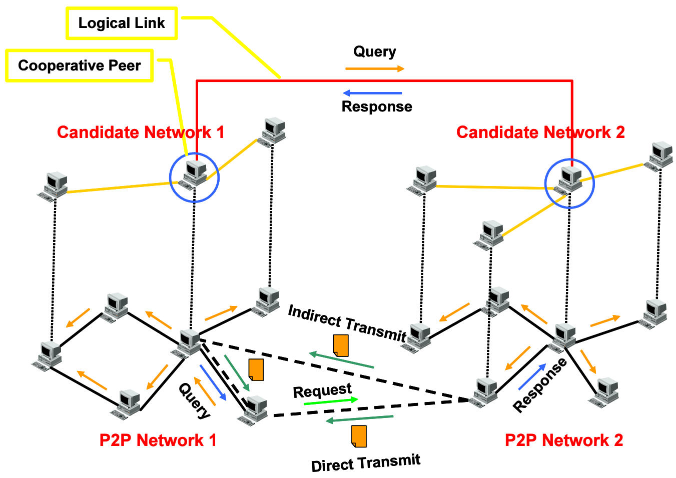

Fig. 2. Cooperation of pure P2P file-sharing networks

the cooperation program independently of others, that is, a peer does not care whether the other peers in the same P2P network will benefit from the cooperation or not. Then, it becomes a candidate peer, i.e., a candidate for cooperative peers. As illustrated in Fig.2, candidate peers in a P2P network construct a candidate network to communicate with each other to select cooperative peers.

A new candidate peer first finds another candidate peer in the same P2P network by flooding a special message over the P2P network or using the i3 network. In the latter case, a new candidate peer registers itself to an i3 service repository by sending a *trigger* message containing a service identifier and its address to the i3 network. On the other hand, candidate peers in a candidate network send *packet* messages containing a service identifier and its address to the i3 network periodically. A new candidate peer receives one of their *packet* messages and establishes a logical link to the candidate peer. After that, the new candidate peer deletes its *trigger* message from the i3 service repository.

For this purpose, candidate peers must have a similar service identifier in the same P2P network but different from those of other P2P networks. In our proposal, a service identifier consists of $l + m + n = 256$ bits. The first l bits are for the cooperation service and common among all cooperation programs. The following m bits correspond to the P2P network. To have the same m bits among candidate peers in the P2P network, we use the IP address of a bootstrapping node. To join a P2P network, a new peer first contacts a bootstrapping node, which should always be available online, to obtain one or more IP addresses of other peers. Since peers in a P2P network know the same bootstrapping node, by applying a hash function to the IP address of the boot strapping node, all candidate peers can have the same network identifier of m bits. We should note here that there is a small possibility that two or more P2P networks have the same m bits identifier. However, we consider that we can avoid the problem without introducing any mediation mechanism. Peers in a P2P network tend to exist close to each other due to a service discovery mechanism of pure P2P applications. Since the i3 network forwards a *packet* message to a node which registers a matching *trigger* message and is close to the sender of the *packet* message, we can expect that a *packet* message is forwarded to another candidate peer of the same P2P network. The last n bits are generated at random. In the i3 network, *inexact matching* is used where the *packet* message has a service identifier matching the longest pattern of bits with the *trigger* message. Therefore, a new candidate peer finds a randomly chosen candidate peer in the same P2P network.

2.2 Selecting Cooperative Peers

Cooperative peers are selected among the candidate peers on receiving a cooperation request. A new cooperation request is generated by a newly joined candidate peer, generated by a candidate peer on its own decision, or sent from other P2P network.

Cooperative peers must be carefully selected to effectively disseminate search messages in P2P networks and distribute the load among peers and networks. It is shown in recent studies [10] that the Internet and many overlay networks have a power-law topology whose degree distribution follows $p(k) \propto k^{-\alpha}$. In [11], it is shown that peers can find files effectively through high-degree peers. It means that by choosing peers with a large number of neighbor peers as cooperative peers, we can expect effective query dissemination. However, high-degree peers are closely connected with each other and thus such selection leads to the concentration of load and causes congestions.

For the efficient and effective message dissemination, we propose a selection method of cooperative peers as follows. First, every candidate peer advertises its degree, i.e., the number of neighbor peers, by flooding a message over a candidate network. Second, each peer ranks candidate peers in descending order of degree. A candidate peer which ranks itself highest advertises a candidacy message to all other candidate peers over a candidate network to become a tentative cooperative peer. On receiving a candidacy message, a candidate peer checks the rank of the tentative cooperative peer in its ranking list. If it is not on the first in the list, a candidate peer sends a conflict message to the tentative cooperative peer. A tentative cooperative peer gives up its candidacy and removes itself from the list on receiving more conflict messages than a predetermined threshold T. The threshold T is introduced to consider the case that a candidate peer, who accidentally missed an advertisement of a tentative cooperative peer, will send a conflict message. Otherwise, a tentative cooperative peer floods a confirmation message with a TTL n in a P2P network. If any cooperative peer already exists within the range, it sends a reject message to the tentative cooperative peer. On receiving a reject message, a tentative cooperative peer gives up its candidacy and advertises its cancellation to the other candidate peers. The tentative cooperative peer is removed from the list and another selection is conducted again.

By this mechanism, cooperative peers are kept apart from each other by more than $n+1$ hops. When a tentative cooperative peer does not receive any reject message in a given time, it finally becomes a cooperative peer. To select two or more cooperative peers, each candidate peer removes a new cooperative peer from the list and repeats the same procedures.

2.3 Finding Other P2P Networks

A newly chosen cooperative peer first finds a candidate peer in other P2P networks by using the i3 network. A cooperative peer sends a trigger message containing a service identifier and its address to the i3 network. The last $m+n$ bits of the service identifier are generated at random, where m bits must be different from its own network identifier.

When a cooperative peer receives a packet message which matches the trigger message by inexact matching, it sends a cooperation request to the candidate peer, i.e., the sender of the packet message, in another P2P network. Next, the selection of a cooperative peer is initiated by the candidate peer in a newly found P2P network. Then, the cooperation request is forwarded from the candidate peer to a new cooperative peer. Finally, a logical link is established between those cooperative peers.

2.4 Decision of Starting Cooperation

Through a logical link established in the preceding step, cooperative peers exchange information to decide whether they cooperate with each other or not. In a biological system, there are varieties of cooperation, coexistence, or, symbiosis, i.e., mutualism, where both species benefit from each other, commensalism, where one species benefits from the other, but the other is unaffected, and parasitism, where one species benefits from the other, but the other suffers. In the case of P2P file-sharing applications, we consider mutualism. However, the decision is still selfish. A peer begins cooperation to enhance and improve its own QoS. A peer maintains an inter-network logical link as far as it considers it is beneficial to itself. When both sides of a logical link consider it is worth connecting, the link is kept. Cooperation is a consequent of selfish behavior of cooperative peers. The decision to start cooperation is made taking into account some criteria, such as the compatibility between P2P file-sharing protocols, the size of P2P networks such as the number of peers and files, and the type of files shared in P2P networks.

When application protocols are different, cooperative peers must convert one protocol into the other. Therefore, it is desirable that protocols are the same or compatible to reduce the load on cooperative peers. When P2P networks are different in their size, peers in a larger P2P network cannot expect the benefit from the cooperation very much. However, the newly introduced load from a smaller cooperative P2P network is considered not much. On the other hand, peers in a smaller P2P network can share and find more files by the cooperation, but they receive a considerable amount of search messages from a larger P2P network. Therefore, cooperative peers must consider the trade-off between the benefit in the application-level QoS and the cost in the increased load by the cooperation. When the type and category of files shared in P2P networks are different, the effect of cooperation is rather small from the viewpoint of the application-level QoS. Therefore, it is desirable that P2P networks sharing similar files such as movies, music, and documents cooperate with each other. A cooperative peer obtains that information and defines priorities to each of them. When the weighted sum is beyond a threshold for both cooperative peers, the cooperation is started. We should note that weight values and the threshold are determined by an application and details of its strategy and policy are left as one of future research topics.

2.5 Relaying Messages and Getting Files

A search message sent from a peer is disseminated in a P2P network by a flooding scheme. When a search message reaches a cooperative peer, it is forwarded to a cooperative peer in another P2P network after protocol conversion is applied if needed. A TTL value of a search message is reduced by one in transmission between cooperative peers. We hereafter call a P2P network from which a search message originated as a guest network and the other as a host network. A cooperative peer in a host network disseminates the search message in the host P2P network by flooding. When there are two or more pairs of cooperative peers, the same search message would be relayed to a host network. To eliminate the duplication, search messages have the same identifier independently of cooperative peers they traverse. Peers in a host network silently discard duplicated search messages with the same identifier.

If a file is found in a host P2P network, a reply message is generated by a provider peer and it reaches a cooperative peer in a host network along a reverse path of the corresponding search message. A cooperative peer in a host network transmits the reply message to a cooperative peer in a guest network after protocol conversion if needed. In the case that a different protocol is used for file retrieval, a cooperative peer in a guest network cashes a reply message and replaces the address of a provider peer with its own address in the reply message. A reply message reaches the source peer of the search message along a reverse path of the search message in a guest P2P network. The searching peer establishes a connection to a provider peer to obtain a file. In the case that a protocol for file retrieval is different, the peer regards a cooperative peer as a provider peer. Then, the cooperative peer retrieves the file from the original provider peer on behalf of the searching peer. Finally, the file is sent to the searching peer. Therefore, peers do not need to recognize such cooperation to receive the benefit of the cooperation.

2.6 Decision of Finishing Cooperation

Cooperation of P2P networks is terminated by disconnection of all logical links established between all pairs of cooperative peers. A logical link is maintained by the soft-state principle. When no message is transmitted through a logical link for a predetermined duration of time S, it is disconnected. In addition, a peer intentionally disconnects a logical link when it considers that it pays too much for the cooperation. As a consequent of the cooperation, which was initiated by a peer itself, the peer helps peers in a cooperating network in finding files by relaying query and response messages. Taking into account the trade-off between the benefit and the cost of the cooperation, a peer decides whether it maintains the link or not. For example, a cooperative peer monitors the number of outgoing messages and that of incoming messages, then compare their ratio to the threshold R, which is determined by an application or a user. We should note here that details of criteria are left as one of future research topics.

3 Simulation Evaluation

In this section, we conduct several preliminary simulation experiments to evaluate our proposed mechanism. To see what happens when two P2P networks cooperate with each other, we consider two cooperative and static P2P networks. Metrics of our evaluation are the reachability of search messages, the number of found files, and the load on peers. The reachability of search messages is the average fraction of the number of peers which a search message reaches among all peers. As the number of reachable peer becomes higher by cooperation, the possibility of successful search also increases. In addition, a searching peer can choose the most preferable, i.e., the fastest or the most reliable, provider peer among the increased number of file-holders. Therefore, with a higher reachability we can expect a higher application-level QoS in P2P file-sharing applications. The number of found files is the average number of files found in P2P networks per search message. The number of found files is equivalent to the number of found file-holders in our experiments. The load on peers is the average sum of search and reply messages which a peer sends, relays, and receives. The load corresponds to the cost which is introduced by cooperation.

3.1 Simulation Environments

We generate two power-law networks of 10,000 peers based on BA model [12] by a topology generator, BRITE [13]. We assume that logical links among peers have infinite capacity and zero latency. We consider static and stable networks where there is no change in their topologies due to joins and leaves of peers. There are F kinds of files in both P2P networks. Their popularity is determined by Zipf distribution of α=1.0. The number of files also follows Zipf distribution of α=1.0, where the number of the least popular file is 1. For example, in a P2P network of 10,000 peers, there are 5,000 kinds of 45,473 files and the number of the most popular file is 5,000. Files are placed on randomly chosen peers.

A search message is generated at a randomly chosen peer for a file determined in accordance with the popularity. It is disseminated by flooding within the range limited by TTL, which ranges from 1 to 10 in our simulation experiments. To

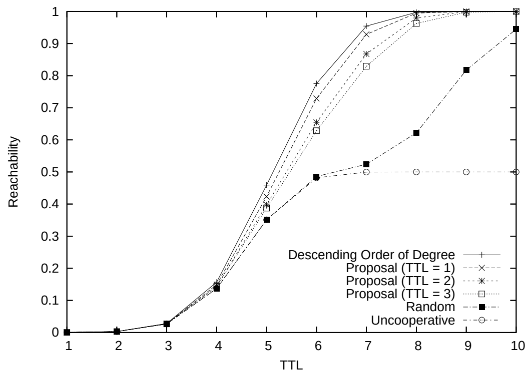

Fig. 3. Relationship between the reachability of search messages and TTL value

keep the distribution of files to follow Zipf, a peer does not retrieve a file in our evaluation.

We change the number of cooperative peers from 1 to 100. Cooperative peers are chosen among all peers, that is, all peers are candidate peers in our simulation experiments. In all cases, 20,000 search messages are generated in P2P networks.

3.2 Evaluation of Reachability of Search Messages

Figures 3 and 4 illustrate the reachability of search messages. In these figures, "Descending Order of Degree" shows the result of the case that cooperative peers are selected in descending order of degree of a peer. "Proposal (TTL = n)" shows the result of the case that cooperative peers are selected by our proposed algorithm. In our proposal, a TTL value of a confirmation message is set at n so that the number of hops among cooperative peers are kept more than $n+1$. "Random" shows the result of the case that cooperative peers are selected at random. "Uncooperative" shows the result of the case that there is no cooperation among P2P networks.

Figure 3 illustrates the relationship between the reachability of search messages and an initial TTL value of a search message where the number of cooperative peers is 10. It is shown that, by the cooperation of P2P networks, search messages reach more peers, and consequently peers can find desired files with a higher probability. In addition, by selecting high-degree peers as cooperative peers preferentially, search messages reach more peers even if a TTL value is small. For example, the reachability of "Uncooperative" and "Random" with TTL of 7 is lower than that of "Descending Order of Degree" and "Proposal" with TTL of 6.

Figure 4 illustrates the relationship between the reachability of search messages and the number of cooperative peers where a TTL value is set at 7. The reachability becomes lower as the number of hops between cooperative peers

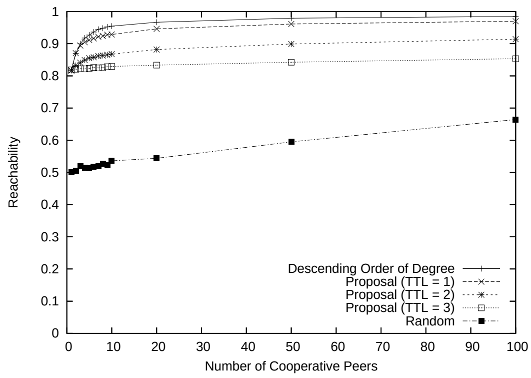

Fig. 4. Relationship between the reachability of search messages and the number of cooperative peers

increases in our proposal. In a power-law network, high-degree peers tend to be located closer, that is, they are connected with each other. Therefore, as n increases, low-degree peers begin to be chosen. Since a low-degree peer cannot disseminate search messages effectively, the number of reachable peers decreases. Furthermore, Fig. 4 shows that in all degree-dependent selection algorithms, the amount of increase in the reachability becomes smaller as the number of cooperative peers increases. Therefore, peers benefit from cooperation with only a few cooperative peers. For example, in the cooperation of two P2P networks of 10,000 peers, about 10 cooperative peers are enough.

3.3 Evaluation of Number of Found Files

Figure 5 illustrates the relationship between the number of found files and the popularity. The number of cooperative peers is 10 and a TTL value is set at 7. It is shown that degree-dependent selection algorithms provide twice the performance of the random selection algorithm and the uncooperative networks. The number of found files of "Random" is almost the same as that of "Uncooperative", that is, the cooperation of P2P networks by randomly chosen cooperative peers does not improve the application-level QoS at all. Since the majority are low-degree peers in a power-law network, a random selection algorithm often chooses low-degree peers as cooperative peers. As a result, the random selection algorithm cannot effectively disseminate search messages in a host network. The reason of step-shaped lines in Fig. 5 is that the number of files, which follows Zipf distribution, takes integer values based on the popularity.

Figure 6 illustrates the cumulative distribution function of the number of found files against the number of hops between a searching peer and file-holders. It is shown that the number of found files within four-hops neighbors is almost the same among algorithms. However, degree-dependent selection algorithms can

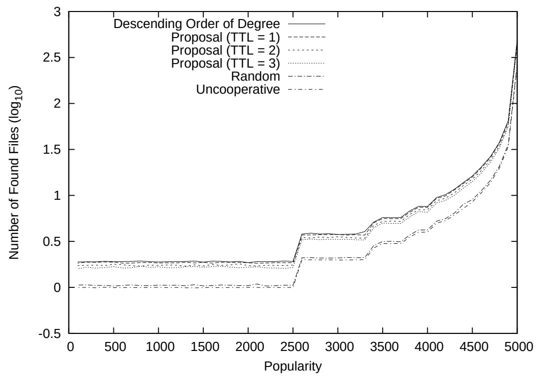

Fig. 5. Relationship between the number of found files and the popularity

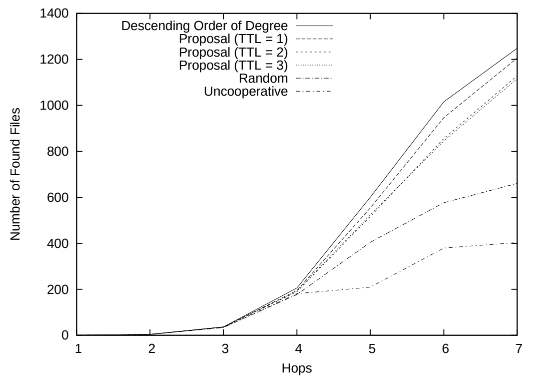

Fig. 6. CDF of the number of found files against the number of hops between a searching peer and file-holders

find more file-holders distant from a searching peer. This comes from the fact that degree-dependent selection algorithms disseminate search messages more effectively in a host network as shown in Fig. 3. However, it takes long time for response messages generated at distant file-holders in a host network to reach a searching peer, since they traverse a reversed path of the corresponding query peer in a logical P2P network. Therefore, to have more file-holders for a higher application-level QoS, it is necessary that a searching peer wait for a longer duration of time.

3.4 Evaluation of Load on Peers

Figures 7 and 8 illustrate the load on peers. Figures 3 and 7, and Figs. 4 and 8 show similar tendency respectively, because the number of search and reply mes-

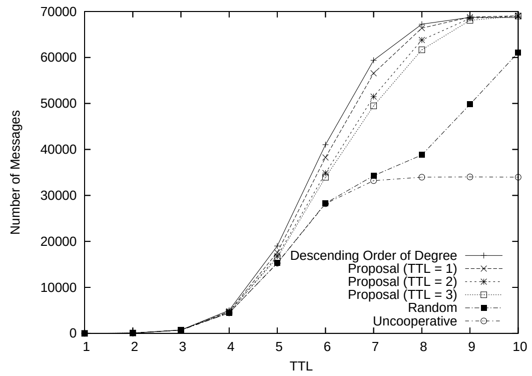

Fig. 7. Relationship between the load on peers and TTL value

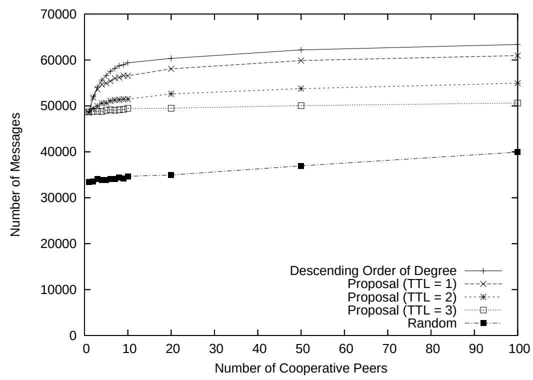

Fig. 8. Relationship between the load on peers and the number of cooperative peers

sages increases in proportional with the number of peers that search messages reach, i.e., the reachability. However, the load increases slower than the reachability against a TTL value, because the number of duplicated search messages becomes small in low-degree peers. For example, the load of "Proposal (TTL = 3)" with TTL of 6 is almost the same as the load of "Uncooperative" with TTL of 7 (Fig. 7), whereas the reachability of the former is higher than that of the latter (Fig. 3).

Figure 9 illustrates the distribution of the number of duplicated search messages that a peer receives. The number of cooperative peers is 10 and a TTL value is set at 7. The duplicated search messages are redundant and lead to the waste of physical network resources and the processing power of peers. In comparison with "Descending Order of Degree", our proposal can reduce the number of duplicated messages especially at peers with a degree smaller than 100. In "Descending Order of Degree", since high-degree peers are selected as

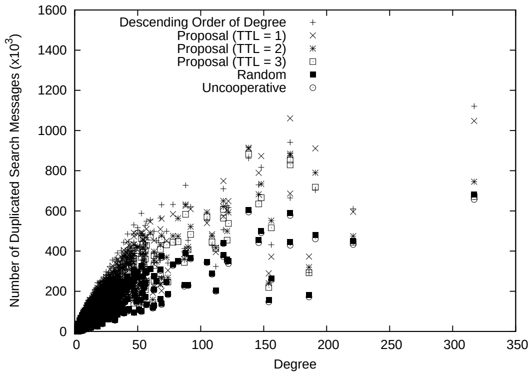

Fig. 9. Distribution of the number of duplicated search messages

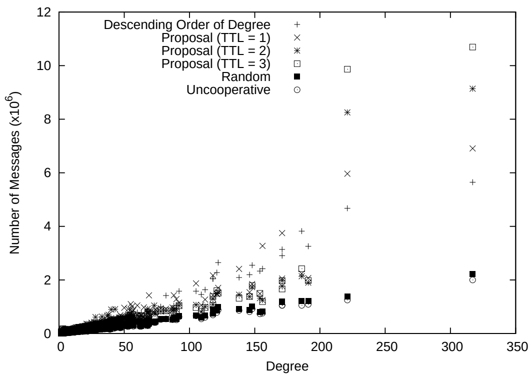

Fig. 10. Distribution of the number of search and reply messages

cooperative peers, search messages via cooperative peers can reach to distant peers. It often happens that search messages are redundant at any peer. On the other hand, in our proposal, cooperative peers are far from each other and consist of both high-degree peers and low-degree peers. Only search messages with a larger TTL value are redundant, as the number of hops between cooperative peers increases.

On the other hand, as Fig. 10 illustrates, the number of search and reply messages including duplicated messages at the highest-degree peers is considerably high in our proposed methods. In addition, as the number of hops between cooperative peers increases, the load on the highest-degree peers increases. A peer with degree 317 and one with degree 221 have the highest degree in each of P2P networks and they are always chosen as cooperative peers. As the number of hops increases, low-degree peers, who disseminate search messages less effectively, are chosen as cooperative peers. Then, the number of peers that search messages originated from a high-degree peer reach increases. As a result, the number of reply messages becomes higher at a high-degree peer with larger n.

4 Conclusions

In this paper, in a context of the overlay network symbiosis, we proposed a mechanism for pure P2P networks of file-sharing applications to cooperate with each other. Through simulation experiments, it was shown that application-level QoS was improved by selecting high-degree peers as cooperative peers in the cooperation of power-law P2P networks. Furthermore, it was shown that by keeping cooperative peers apart from each other, the load on the P2P network was reduced, but a few cooperative peers were burden with heavy load.

As future research works, first we consider a mechanism to distribute the load among cooperative peers while keeping the high reachability. We also investigate behaviors of cooperation among dynamic P2P networks, which change their topology as consequences of cooperation. Furthermore, we should evaluate influences of cooperation to a physical network.

Acknowledgements

This research was supported in part by "New Information Technologies for Building a Networked Symbiosis Environment" in The 21st Century Center of Excellence Program of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

1. L. Qiu, Y. R. Yang, Y. Zhang, and S. Shenker, "On Selfish Routing in Internet-Like Environments," in *Proceedings of ACM SIGCOMM Conference 2003*, pp. 151–162, Aug. 2003.
2. M. Seshadri and R. H. Katz, "Dynamics of Simultaneous Overlay Network Routing," in *Technical Report of Electrical Engineering and Computer Sciences (EECS), University of California Berkeley (UCB), UCB/CSD-03-1291*, Nov. 2003.
3. M. Kwon and S. Fahmy, "Toward Cooperative Inter-overlay Networking," in *Poster in the 11th IEEE International Conference on Network Protocols (ICNP)*, Nov. 2003.
4. I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana, "Internet Indirection Infrastructure," in *Proceedings of ACM SIGCOMM Conference 2002*, pp. 73–88, Aug. 2002.
5. A. Nakao, L. Peterson, and A. Bavier, "A Routing Underlay for Overlay Networks," in *Proceedings of ACM SIGCOMM Conference 2003*, pp. 11–18, Aug. 2003.
6. D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient Overlay Networks," in *Proceedings of the 18th ACM Symposium on Operating Systems Principles (SOSP)*, Oct. 2001.

7. T. Yomo, W.-Z. Xu, and I. Urabe, "Mathematical Model Allowing the Coexistence of Closely Related Competitors at the Initial Stage of Evolution," in *Researches on Population Ecology, vol.38, no.2*, pp. 239–247, 1996.
8. N. Wakamiya and M. Murata, "Toward Overlay Network Symbiosis," in *Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing (P2P2005)*, Aug. 2005.
9. R. Schollmeier and G. Schollmeier, "Why Peer-to-Peer (P2P) Does Scale: An Analysis of P2P Traffic Patterns," in *Proceedings of the Second IEEE International Conference on Peer-to-Peer Computing (P2P2002)*, Sept. 2002.
10. M. E. J. Newman, "The Structure and Function of Complex Networks," in *SIAM Review, vol.45, no.2*, pp. 167–256, 2003.
11. L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman, "Search in Power-law Networks," in *Physical Review E, vol.64, 046135*, Sept. 2001.
12. A. L. Barabasi and R. Albert, "Emergence of Scaling in Random Networks," in *Science, vol.286*, pp. 509–512, Oct. 1999.
13. A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An Approach to Universal Topology Generation," in *Proceedings of the International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication System (MASCOTS'01). available at http://www.cs.bu.edu/brite/*, 2001.