Journal of Information Processing Vol.22 No.1 30–39 (Jan. 2014)

[DOI: 10.2197/ipsjjip.22.30]

Regular Paper

Efficient Information Dissemination and Reputation Aggregation for User Centric Media Utilizing the Circular Board Method

Yoshio Sakurauchi[1,a) Hideyuki Takada[2,b)]

Received: April 4, 2013, Accepted: October 9, 2013

Abstract: Recently user-generated content (UGC) has become major content of the Web and one of the most important factors of UGC is who has generated it. Even if the same information is disseminated, its credibility is different according to its author. Typically, authors are characterized by reputation systems. Although cloud computing enables both information dissemination and reputation aggregation with scalability, it is better to minimize the use of clouds due to cost problems. In this paper, we propose to apply the circular board method based on Chord to user centric media to disseminate information and aggregate reputation efficiently in a P2P manner. Its ring topology makes it possible to effectively collect the reputation from users at the same time when each piece of UGC passes through user terminals. The results of simulations reveal the feasibility of P2P information dissemination and reputation aggregation and provide insights about trade-offs between network resource consumed and time required for information dissemination and reputation aggregation.

Keywords: information dissemination, reputation aggregation, UGC

1. Introduction

Recently user-generated content (UGC) has become major content of the Web [1] and one of the most important factors of UGC is who has generated it. Through user centric media such as Twitter, Facebook, and YouTube, users are producing and consuming UGC actively. UGC is not only useful in daily life but also in serious incidents and natural disasters. In some cases, users can get information about them earlier than traditional media like TV and radio [2]. Several user centric media have reputation systems like a voting system to characterize UGC and its author. Because even if the same information is disseminated, its credibility is different according to its author.

Although cloud computing enables both information dissemination and reputation aggregation with availability and scalability, it is better to minimize the use of clouds due to cost problems and some other reasons, such as benefiting from P2P-based systems. Generally UGC is treated by the conventional client-server architecture. This means that, with the help of cloud computing, there is no limit of scalability in a sense if anyone can provide enough money. Meanwhile, P2P-based information sharing systems have been used to realize scalable, fault-tolerant and sometimes anonymous information sharing [3]. In addition, P2P-based systems can exploit local networks of user terminals such as LAN and MANET. Some academic and commercial projects have tried to mix the two paradigms to take advantages of both [4]. For instance, the cloud can be used just for bootstrapping P2P networks because of its availability.

In this paper, we propose to apply the circular board method based on Chord [5] to user centric media to disseminate information and aggregate reputation at the same time efficiently in a P2P manner. The circular board is a part of Japanese culture and used in communities to share information. If someone wants to disseminate information to their community, he/she puts the information on a (physical) circular board and the community members pass around the circular board in order (physically). The community members may take some actions such as signing, making comments, and so on when they get the circular board and before they pass it to the next member. This circular board method would be also proper for (digital) user centric media especially in a P2P manner. We employed Chord to determine the order of passing the circular board because Chord is the most common algorithm to organize a P2P cyclic topology. Its cyclic topology makes it possible to effectively collect the reputation from users at the same time when each piece of UGC passes through user terminals. The results of simulations provide insights about trade-offs between network resource consumed and time required for information dissemination and reputation aggregation.

The rest of the paper is organized as follows. Section 2 provides a survey of related work. In Section 3, the system model is proposed. The simulation results are shown in Section 4 and discussions are made in Section 5. Finally, Section 6 concludes

[1] Graduate School of Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 525–8577, Japan
[2] College of Information Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 525–8577, Japan
[a] y_sakurauchi@cm.is.ritsumei.ac.jp
[b] htakada@cs.ritsumei.ac.jp

the paper and gives an outlook on the future work.

2. Related Work

Several system models have been proposed to disseminate information to interested users. While the circular board model can achieve information dissemination and reputation aggregation simultaneously, these models only set a goal to information dissemination. The same thing can be said about reputation aggregation algorithms.

2.1 Information Dissemination

The publish/subscribe model, the gossip model, and the P2P streaming model are three widely used system models for information dissemination.

2.1.1 Publish/Subscribe Model

The publish/subscribe model divides users into publishers and subscribers. Publishers publish information without specific receivers. Subscribers can subscribe to any publishers and get information published by them.

Twitter is a typical example of the publish/subscribe model and one of the most popular microblogging services. Users can send posts of up to 140 characters, called "tweets," and subscribe to other users' tweets. This action to subscribe is called "follow" and subscribing users are called "followers." Generally Twitter is also regarded as an online social networking service because of these following-followed relationships. Users of Twitter are not limited to only individuals but also organizations such as companies and universities [6]. Automatic programs are also handled as users. That kind of programs post popular tweets, tweets in specific fields, and so on. In addition, Twitter has a function, called "retweet," which enables users to diffuse information they credited to their followers. These facts mean that users can get information from other users who they are not directly following.

Some publish/subscribe services are built on top of a distributed hash table (DHT) unlike Twitter which is built on the cloud. There are several ways to maintain or generate dissemination tree on top of DHT [7]. Ferry [8] proposes an architecture that extensively yet wisely exploits the underlying DHT overlay structure to build an efficient and scalable platform for content-based publish/subscribe services. Moreover, some approaches which enable subscribers to get information from unforeseen publishers have been tried to provide users more opportunity to get unknown information. For instance, keyword-based content dissemination has been tried instead of publisher-based content dissemination [9].

2.1.2 Gossip Model

The gossip model mimics word of mouth in the real world. Information is disseminated by users who think it is worth spreading. In case of using wireless ad-hoc networks between mobile terminals, users who did not encounter anyone cannot obtain information. Using online social networks [10], on the other hand, users can get information wherever they are in the real world.

The gossip model can be also regarded as an epidemic model especially when gossip systems work without users' actions such as sharing with someone. In this case, the peer sampling is a very important matter. A peer sampling mechanism determines which peers should share information with which peers. Some gossip peer sampling mechanisms postulate the cloud and physical network [11]. Reference [4] has proposed a hybrid architecture of the cloud and P2P based on Cyclon [12]. It has also shown that the economic cost can be reduced effectively in commercial services, the hourly News Update podcast from CNN and the Dilbert's comic strips.

2.1.3 P2P Streaming Model

The P2P streaming model uses the tree and/or mesh topology to distribute information among a large number of users [13]. Information is delivered from a server to clients in a continuous fashion. This model has been used in commercial services such as P2P TV and displayed its scalability [14].

Basically this model assumes that information is delivered from a few users to a large number of other users. This means that, if a large number of users tries to disseminate information simultaneously, it is very hard to handle them and sometimes it will take them out of service.

2.2 Reputation Aggregation

Needless to say, if web servers can be used, it is easy to aggregate reputations from users. However it is not easy in P2P-based systems because which peer should maintain aggregated reputations is a difficult problem.

In GossipTrust [15], each peer has local scores representing reputations about all other peers and share the local scores with randomly selected peers. PTrust [16] has tried to combat malicious peers by enabling peers to send occasional messages to lower the global scores of malicious peers as soon as they make mischief. These algorithms are effective only for reputation aggregation and do not have a function to disseminate information. The circular board method, on the other hand, can work as reputation aggregation mechanism at the same time with information dissemination mechanism. The details are described in Section 3.2.2.

Meanwhile, Aggregation Skip Graph [17] has enabled efficient execution of range query for aggregation in Skip Graph [18] which is a kind of distributed data structure based on skip lists. In particular, minimum and maximum values can be computed with fewer messages as the query range becomes wider. Range query in Skip Graph can also be used to disseminate information. We have a discussion regarding this point in Section 5.5.

3. System Model

In this section, we propose the circular board method based on Chord and provide an example of how it works to categorize transmissions of circular boards.

3.1 Layered Model of User Centric Media

A model of user centric media employed in this research is illustrated in **Fig. 1**. The architecture consists of three layers. The small circles indicate user terminals such as computers and smartphones. The role of each layer is described below.

3.1.1 Cluster Layer

The cluster layer manages user groups. Users are clustered in accordance with their interests. Users may belong to multiple

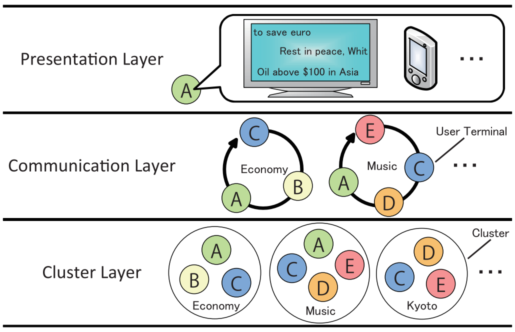


Fig. 1 A three-layer model of user centric media.

groups at the same time. The large circles surrounding user terminals indicate clusters of user terminals. User terminals marked by the same label represent the same user terminal.

We do not feature how these clusters are formed because it is not the central topic of this paper to cluster users; users shall be clustered in some fashion. There are many well-known soft clustering methods [19]. Instead of automatic clustering, users may create, join and leave a cluster manually.

Whenever users join/leave a cluster, its event is reported to the communication layer so that their terminals can be included in or excluded from the target of information dissemination and reputation aggregation. It should be emphasized that user management mechanisms in the cluster layer are also responsible for detecting an unexpected withdrawal of user terminals by periodic check or other methods.

3.1.2 Communication Layer

The communication layer controls how to disseminate information and aggregate reputations in each of clusters. This layer is the heart of this paper. We propose to use the circular board method based on Chord, which can achieve information dissemination and reputation aggregation simultaneously, in this layer. Further details are provided in the following sections.

3.1.3 Presentation Layer

The presentation layer provides methods to present information to users. Presentation methods would vary according to the difference of types of user terminals such as smartphones, tablets and desktop computers. Although we do not strictly define how disseminated information must be presented, presentation methods are expected to express which clusters information came from because users might belong to several clusters and information is delivered from each of the clusters. Even if the same information is posted, the information will be public or foreclosed by the difference in clusters. This means that credibility of information depends on not only who posted it but also which clusters it came from. For an extreme example, if a piece of information which says “Japan did default” is disseminated in a joking cluster, most users will never believe it and just skip it. In contrast, if it is disseminated in an economic cluster, users will be surprised and try to check if it is true or not in some way. How to express multiple properties of a lot of information could be an additional research issue.

3.2 Chord-based Circular Board Method

3.2.1 Chord

In the communication layer, user terminals (hereinafter called peers) in the same cluster are connected with each other by Chord. Chord is a DHT algorithm and organizes a ring topology of peers. The simple ring topology is realized by only the successor list which each peer has. Successors are peers which come after a peer in the ring topology. If the length of a successor list is 1, a peer only knows the next peer. If the length is 2, a peer knows the next peer and the peer after the next peer. The same goes for the following. Peers maintained by Chord also have special shortcuts to other peers called the finger table which enable a peer to communicate with distant peers. Successor lists and finger tables can be generated by the typical algorithm of Chord easily [5].

Besides, we assume that reorganization of Chord only happens when users join or leave the cluster. In other words, even if an online/offline status of peers has changed, reorganization of Chord does not happen as long as the user belongs to the cluster. Connection failures—peers tried and failed to communicate with offline peers—are ordinary and admissible in the proposed system. When a peer left permanently from Chord network for some reason, its situation is detected by user management mechanisms in the cluster layer and reorganization of Chord will be prompted based on the notice from the cluster layer.

3.2.2 Circular Board Method

In the proposed system, a piece of information (UGC) is treated by a circular board. The basic format of circular boards is as follows.

- Board ID
- Information content (text, image, and so on)
- Author information
  - General profile (user name and etc.)
  - Latest reputation
  - Peer ID
- Collection of reputations

A circular board has a board ID and contains UGC, its author information and collection of reputations. The author information provides a general profile of the author and its latest reputation. A latest reputation would be presented by numerical scores or user ranks so that other users can estimate credibility of information. The author’s peer ID is used to determine the pathway of the circular board. The collection of reputations contains reputations from other peers which the circular board has passed through.

When a user generates information, a circular board starts out from and arrives back in the user’s peer. During that time, the circular board flows among other peers along the topology of Chord and other users can watch and/or listen to them. The ring topology makes it possible to collect the reputation from other users at the same time when the circular board passes through other peers. Note that users do not have to evaluate all UGC.

The algorithm for consumer peers, the peers which receive circular boards, is indicated in **Algorithm 1**. Once a circular board arrives from other peers, onReceive() is called. In the procedure, the circular board which is already received is not processed but one which is not received before is passed to notice() so that presentation methods in the presentation layer can provide it to users.

Algorithm 1 Procedures for consumer peers

```
procedure onReceive(info)
    if info is not already received then
        notice(info)
        receivers ← getReceiver(info)
        for i ← 0, receivers.length do at prescribed intervals
            send(receivers[i], info)
        end for
    end if
end procedure

procedure evaluateInfo(info, eval)
    info.add(eval)
end procedure

procedure getReceiver(info)
    successor ← peers in the successor list of Chord
    if the finger table is enabled then
        finger ← peers in the finger table of Chord
        receivers ← successor + finger
    else
        receivers ← successor
    end if
    for i ← 0, receivers.length do
        if receivers[i] jumps over info.peer then
            Remove receivers[i] from receivers
        end if
    end for
    sort(receivers)                    ▷ Farthest first in principle
    return receivers
end procedure
```

Algorithm 2 Procedures for producer peers

```
procedure postInfo(info)
    receivers ← getReceiver()
    for i ← 0, receivers.length do at prescribed intervals
        send(receivers[i], info)
    end for
end procedure

procedure getReceiver( )
    successor ← peers in the successor list of Chord
    if the finger table is enabled then
        finger ← peers in the finger table of Chord
        receivers ← successor + finger
    else
        receivers ← successor
    end if
    sort(receivers)                    ▷ Farthest first in principle
    return receivers
end procedure

procedure onReceive(info)
    tallyReputation(info)
end procedure
```

Users may give a reputation to the UGC in some cases using evaluateInfo(); the parameter will be a score or comments. Subsequently, the peer determines receivers using the peer ID of the circular board. getReceiver() returns a receiver list based on the settings, whether the finger table is enabled or disabled and how to sort receivers. The peer ID is used not to transmit the circular board to receivers across the producer peer (the author's peer). Once receiver peers are determined, the peer starts disseminating the circular board to them at prescribed intervals. This interval will vary according to applied applications. Note that user evaluation of UGC can occur simultaneously when the dissemination is ongoing. In that case, the reputation is only transmitted to the remaining receivers which have not received the circular board yet.

The algorithm for producer peers, the peers which generate circular boards, is indicated in **Algorithm 2**. Once a user produces information, the peer determines receivers based on the settings, whether the finger table is enabled or disabled and how to sort receivers, and starts disseminating the circular board to them at prescribed intervals. When the circular board arrives back in the peer, the reputations from users are tallied using tallyReputation(), which will calculate a score and so on. Note that a circular board may arrive back in the producer peer with some replicas which are duplicated in the dissemination process; one peer transmits a circular board to multiple receivers in the dissemination process and each transmitted circular board can be regarded as replicas of the original circular board. In that case, all circular boards containing the same board ID are merged to one in the tallying process.

3.2.3 Example and Effective/Ineffective Transmission

An example of information dissemination and reputation collection by the proposed method is illustrated in **Fig. 2**. The small circles indicate user terminals. In the example, the length of a successor list is 3 and the finger table is not enabled. The user of peer S produces information at $t = 0$ and the peer S starts disseminating the circular board at $t = 1$. Every peer tries to transmit the circular board to the farthest peer in the receiver list for each interval. During the transmission phase, reputations are collected as attachments of the circular board if users give evaluation on the information. When the circular board comes back to the peer S, the collected reputations are tallied.

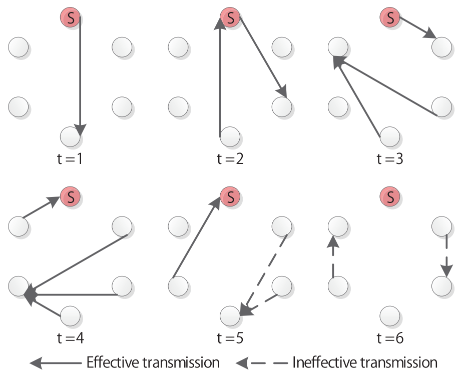


Fig. 2 A pathway of circular board.

Transmissions of circular boards can be categorized into two types, the effective transmission and the ineffective transmission. The effective transmission is defined as transmissions to uninformed peers or transmissions which finally lead to the producer peer. The former contributes to information dissemination and the latter contributes to reputation. The other transmissions are defined as the ineffective transmission. Note that if some effective transmissions (in a fail-proof condition) are failed for any reason, those transmissions are treated as just the connection failure and some ineffective transmissions (in a fail-proof condition) would be treated as the effective transmission.

4. Simulation

First we define indices which are used to observe and compare the network efficiency for information dissemination and reputation aggregation. Through the use of the indices, the relationship between the connection failure rate and other settings is investigated so that the proposed system can be applied to a practical application with proper settings; the efficiency are largely dependent on an average connection failure rate which will be estimated from features of the underlying networks and/or user behaviors. Since it is difficult to prepare real environments with different connection failure rates, we have implemented a simulator.

4.1 Indices of Efficiency

The network efficiency at the time interval t can be computed as below.

$$E(t) = \frac{w_{ip} \cdot ip_t + w_{tr} \cdot tr_t}{np} \cdot \frac{w_{et} \cdot et_t}{w_{et} \cdot et_t + w_{it} \cdot it_t + w_{cf} \cdot cf_t} \quad (1)$$

np is the number of peers in the cluster. The meanings of other variables are ip_t: the number of informed peers (peers which received a circular board), tr_t: the number of tallied reputations, et_t: the number of effective transmissions, it_t: the number of ineffective transmissions, and cf_t: the number of connection failures at the time interval t respectively. If the terms are expressed without the subscript t like ip hereafter, that means the final number of each term like the total number of finally informed peers. w_{ip}, w_{tr}, w_{et}, w_{it} and w_{cf} represent a weight for ip, tr, et, it and cf respectively. The early part of the formula represents the coverage of informed peers and tallied reputations. The more peers are informed and reputations are tallied, the higher the network efficiency is. The latter part of the formula means that it is better to decrease the number of ineffective transmissions and connection failures.

The network efficiency is only comparable in the same sequence of information dissemination and reputation aggregation, because if any settings such as the length of a successor list are changed, the number of effective transmissions changes, for instance. To figure out which setting is the most efficient in total, we propose the relative total efficiency as below.

$$E = C \cdot E_{net} \cdot E_{time} \quad (2)$$

Each term is formulated as below.

$$C = \frac{w_{ip} \cdot ip + w_{tr} \cdot tr}{np} \quad (3)$$

C is the coverage of informed peers and tallied reputations.

$$E_{net} = w_{et}\left(1 - \frac{et}{et_{\max} + 1}\right) + w_{it}\left(1 - \frac{it}{it_{\max} + 1}\right) + w_{cf}\left(1 - \frac{cf}{cf_{\max} + 1}\right) \quad (4)$$

E_{net} is the relative total network efficiency. $et_{\max}$, $it_{\max}$ and $cf_{\max}$ represent the maximum number of et, it and cf respectively from results which need to be compared. This formula means that the fewer the number any transmissions becomes, the higher the network efficiency is.

$$E_{time} = 1 - \frac{tm}{tm_{\max} + 1} \quad (5)$$

Likewise $tm_{\max}$ represents the maximum number of tm from results which need to be compared. This formula means that it is better if time required for information dissemination and reputation aggregation gets faster.

4.2 Simulation Setting

Settings for the simulation are as follows.

Number of peers

The simulations have been done in the case when the number of peers is 2^{10}, 2^{13}, and 2^{16}. One of the peers works as a producer peer and others work as consumer peers. Although the number of peers does not change in a simulation dynamically, the offline status of peers is treated as the connection failure as mentioned later.

Chord

The length of a successor list ranges from 1 to 10. When the finger table is disabled, only the successor list is used to determine receivers. If the finger table is enabled, peers in a finger table are also treated as receivers. Hence, some peers in a successor list may overlap with peers in a finger table. In that case, peers in the finger table which also appear in the successor list are not used. Besides, it is assumed that peers are uniformly distributed in the hash space of Chord (160 bits) for making a finger table operate efficiently; Chord has a technique called the virtual peer to achieve this assumption [5].

Connection Failure Rate

The connection failure rate ranges from 0% to 90%. This parameter directly affects the probability of connection failure when each peer tries any transmission. The assumed causes of the connection failure contain the offline status of peers and network failures.

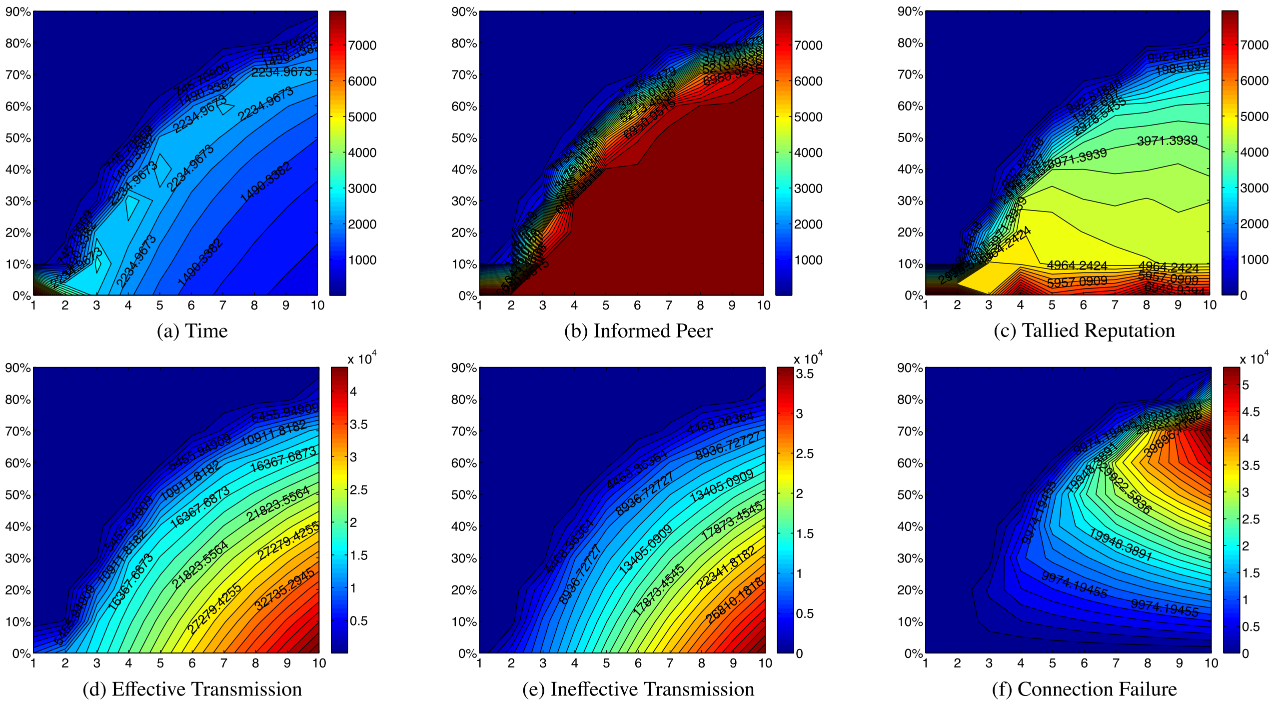
4.3 Simulation Scenario

At the beginning of a simulation, peers are generated and the ring topology of Chord is organized according to the settings. Then a piece of information starts flowing from the producer peer and an elapsed-time counter starts. When each peer receives a circular board, the information is given an evaluation immediately within an interval. In the transmission phase, each peer tries to transmit the circular board in order, from the farthest peer in candidates, to receivers which have not been transmitted the circular board by the peer yet. The reason for the farthest-first transmission is because the results of preliminary experiments show that the farthest-first transmission is the most suitable from the view point of time required for information dissemination and reputation aggregation. Additionally, for each unit time (interval), the number of informed peers, tallied reputations, effective transmissions, ineffective transmissions, and connection failures are recorded.

Simulations stop when every peer finishes all transmissions for its receivers or becomes static in some conditions. Each combination of the settings is repeated 100 times and the averages are calculated.

4.4 Result

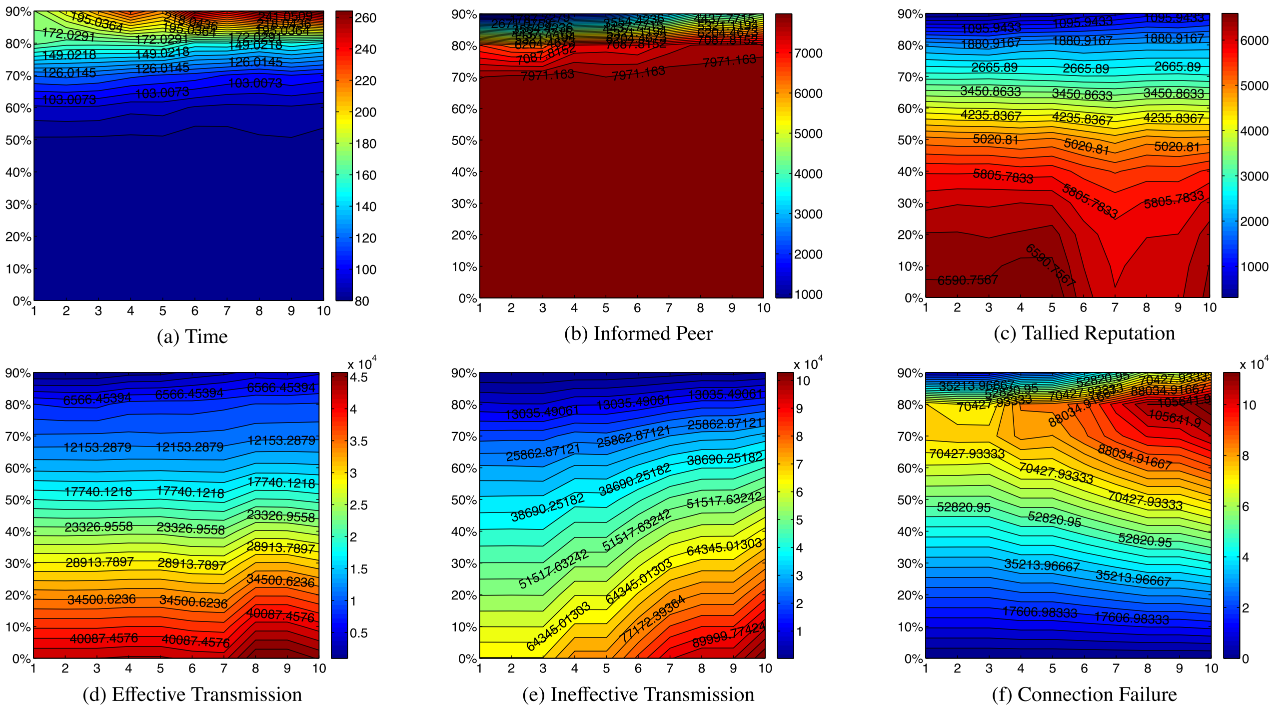
The results of the simulation for 2^{13} peers are illustrated in **Fig. 3** and **Fig. 4**. The finger table has been disabled in Fig. 3 and enabled in Fig. 4. In each contour plot, the horizontal axis, the vertical axis and the contour represent the length of a successor list, the connection failure rate, and the counts for each index respectively.



(a) Time (b) Informed Peer (c) Tallied Reputation

(d) Effective Transmission (e) Ineffective Transmission (f) Connection Failure

Fig. 3 Successor only - 2^{13} peers.



(a) Time (b) Informed Peer (c) Tallied Reputation

(d) Effective Transmission (e) Ineffective Transmission (f) Connection Failure

Fig. 4 Successor with finger table - 2^{13} peers.

Focusing on simulation time, although the upper left part of Fig. 3 (a) indicates that the simulations have finished in a short time under a condition where the length of a successor list is short and the connection failure rate is high, this indication does not mean that information dissemination and reputation aggregation have finished in a short time as shown in Fig. 3 (b) and (c). This may be because circular boards cannot be transmitted to the receivers during the early stage and the all peers become static under the condition.

Figure 3 (b) and (c) show that the number of tallied reputations is influenced strongly than the number of informed peers by the connection failure rate. In contrast, Fig. 3 (d) and (e) indicate that the distribution of the effective transmission count is similar to that of the ineffective transmission count in large. The number of connection failures is simply controlled by the connection failure rate as seen in Fig. 3 (f).

In Fig. 4 (c), the number of tallied reputations decreases when the connection failure rate is low and the length of a successor list is around 7. This type of phenomenon is also seen when the number of peers is 2^{10} and 2^{16}. As the number of peers increases, the

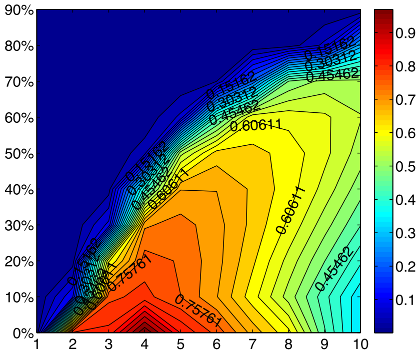


Fig. 5 Total efficiency - successor only - 2^{13} peers.

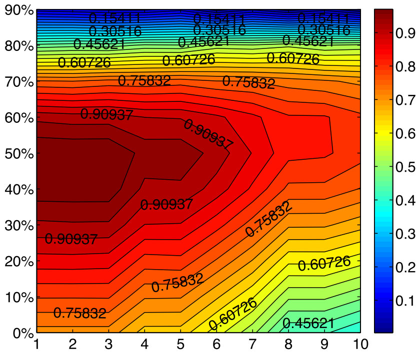


Fig. 6 Total efficiency - successor with finger table - 2^{13} peers.

phenomenon area moves in a direction from the right to the left. This phenomenon may be caused by a combination of the number of peers and the finger table which organizes a specific topology. Because this phenomenon is not seen when the connection failure rate is high; a high connection failure rate destroys a topology of peers virtually.

In Fig. 4 (e), the lower right part is higher than the lower left part since the number of trying transmissions increases as the length of a successor list is getting long when the connection failure rate is the same. In Fig. 4 (f), the same holds true for the upper left part which is lower than the upper right part.

In comparison with Fig. 3, Fig. 4 shows that although the finger table enables very fast information dissemination and information aggregation and provides a high tolerance for the connection failure, it requires a lot of transmissions which are several times higher.

The relative total efficiency for Fig. 3 and Fig. 4 is calculated as **Fig. 5** and **Fig. 6** respectively. Again, the finger table has been disabled in Fig. 5 and enabled in Fig. 6. A weight for the number of connection failures is fixed to 0.5 and the others are fixed to 1. These weights are chosen under the assumption that a cost of the connection failure is about half of the effective/ineffective transmission and the other concerns have the same level of importance. Moreover, in order to make the values of efficiency easier to understand, each value is normalized under the condition that a maximum becomes 1.

When the finger table is disabled, the efficiency becomes a maximum under the condition that the connection failure rate is 0% and the length of a successor list is 4 according to Fig. 5. This tendency does not change even if the number of peers is 2^{10} or

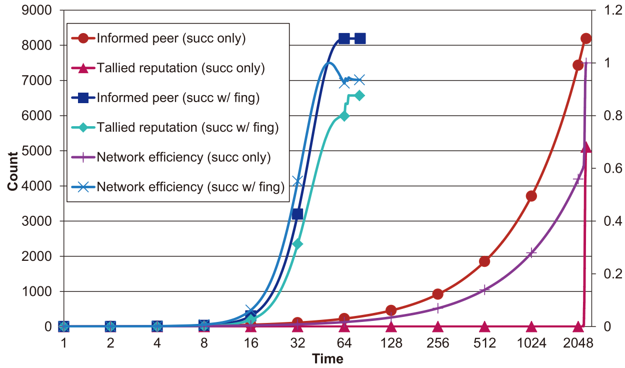


Fig. 7 Effect of the finger table.

2^{16}. In contrast, when the finger table is enabled, a maximum area as can be seen in Fig. 6 has moved in a direction from the bottom to the top with the increase of the number of peers.

Although the other results are omitted due to limitations of space, characteristic results have been touched as stated above. The results which have not been touched mean that they have denoted the same tendency of Fig. 3 through Fig. 6.

5. Discussion

5.1 Employment in Real Environment

According to a survey [20], from January 15th to March 19th in 2013, a success rate of packet communication between smartphones and web servers is about 96% in Japan. In other words, a success rate of packet communication between a pair of smartphones will be about 92% which can be calculated by 96% × 96%. This corresponds to the connection failure rate of 8%.

If the Japanese mobile telephone network is used as the underlying network of the proposed system, with 2^{13} peers, a proper length of a successor list is 4 when the finger table is disabled or 3 when the finger table is enabled. **Figure 7** represents the number of informed peers, tallied reputations, and the network efficiency for each of the conditions. To calculate the network efficiency, a weight for the number of connection failures is fixed to 0.5 and the others are fixed to 1. Note that the network efficiency is not comparable with a different sequence as mentioned before and is normalized under the condition that a maximum becomes 1. In addition, the horizontal axis for time is logarithmic.

When the finger table is enabled, the number of informed peers and tallied reputations increases in a similar way as time goes along. On the other hand, when the finger table is disabled, although the number of informed peers increases as time goes along, the number of tallied reputations does not increase until just before the last part of the graph.

From the view point of the network efficiency, an apparent peak can be seen when the finger table is enabled. This means that many ineffective transmissions and/or connection failures occur in the last part of the simulations. In contrast, such peak cannot be seen when the finger table is disabled.

Obviously the finger table makes time required for information dissemination and reputation aggregation much shorter. For instance, if a unit time (interval) is 10 minutes, it takes 13.5 hours with the finger table and 384 hours without the finger table to disseminate information and aggregate reputations in 2^{13} peers on average. From a viewpoint of network resources, the finger table causes 3.38 times the transmissions including effective transmissions, ineffective transmissions and connection failures. It will depend on each operation policy of individual services whether this increase in network cost is allowable or not. In Twitter, 75% of retweets (the user operation to diffuse information to other users) occurs within one day and about 10% take place a month later [21]. Retweeted information can be regarded as meaningful information at that time. In other words, if our method is applied to a Twitter-like service, most information should be diffused within one day. This could be a trade-off between the required time and network resource which is consumed.

It should be also mentioned about the feasibility of the proposed method with poor terminals such as smartphones. One of the possible implementations would use WAP Push API [22] which is typically used by service providers to push information to consumers like e-mail and flash news. Using this API, smartphones can send/receive information through HTTP POST connections. According to a performance survey [23], the realistic maximum number of HTTP connections is 17 in Android 4.x terminals, 23 in iPhone 5, and more than 60 in Windows Phone 7/8 terminals. In our method, a peer transmits a circular board in series not in parallel as seen in Algorithm 1 lines 5-7. This means that only one HTTP connection is required for each user cluster to pass around circular boards. Consequently, we believe that our method works practically with smartphones even if a user belongs to multiple—but not so many—clusters under the reasonable data size of information and the properly selected unit time (interval). In regard to the network cost of maintaining Chord, the influence of a joining/leaving terminal is limited to several associated terminals due to a feature of Chord [5]. Moreover, reorganization of Chord only happens when users join or leave the cluster and do not happen when an online/offline status of peers just has changed as mentioned in Section 3.2.1. For these reasons, although it is hard to estimate how often reorganization of Chord is required, we predict that its frequency is very low compared with the transmission of circular boards and smartphones can handle it adequately.

5.2 Collection of Reputations

In the simulation, it is assumed that all users give evaluation on the information immediately within an interval when each peer receives a circular board. This assumption would not be realistic.

Table 1 shows the thresholds of required reputations to guarantee a level of statistical significance. These thresholds are calculated under the common assumption in statistics that the confidence level is 99% and the population proportion is 50%. If the number of tallied reputations exceeds the thresholds, the reputations can be treated as a unified reputation of all users with each confidence level.

As the number of peers increases, the number of required reputations also increases, however, its amount of increase gets smaller. In other words, the number of required reputations converges to a particular value with the same maximum error rate. For example, if a maximum error rate is 5%, the number of required reputations converges to about 664.

Table 1 Number of required reputations.

		# of peers		
		2^{10}	2^{13}	2^{16}
Maximum error	0%	1024.00	8192.00	65536.00
	1%	964.70	5489.81	13271.33
	2%	821.90	2759.30	3911.98
	3%	659.25	1508.67	1798.29
	4%	516.23	922.99	1023.83
	5%	403.65	615.69	658.96

How many users evaluate information and how much accuracy is necessary depend on an application. For instance, in the example of Section 5.1, up to 6572.81 reputations can be tallied with the finger table and 5101.68 without the finger table on average. If 25% of users evaluate within an interval, these numbers turn out to be about 1643 and 1275 which guarantee a maximum error rate of 3% and 4% respectively. This could be another trade-off.

5.3 Surrogate for Producer Peer

Although the offline status of producer peers is treated as the connection failure in the simulation, in actual situations, some producer peers may go offline for a very extended period of time after the users generate information. One simple solution is to prepare surrogates such as super peers and web servers. If transmissions to producer peers fail, the transmitters can simply transfer them to the surrogates with ID of the producer peer. When the producer peers get back online, they can get their reputations from the surrogates.

The super peer can be elected from peers based on the performance of CPU, the capacity of network and duration time of online. Multiple super peers would be required to deal with network failures and sudden defection of some super peers. Alternatively the cloud can be used as mention in the introduction. It can also solve the bootstrap problem, how to find an existing peer, and enable the user management like expulsion of immoral users easily which often cause trouble in P2P-based systems.

5.4 Content Distribution Network

In some cases, it is not realistic to transmit vast amounts of information like high quality videos directly by information dissemination systems including the proposed system in this paper. For handling that kind of information, a combination of a content delivery network (CDN) and information dissemination systems is one of the most potent solutions. CDN enables a large number of consumers to download vast amounts of information simultaneously [24].

For instance, a pair of an information summary and a URI link to the body of the information may be disseminated and the body may be located in CDN-assisted hosts. If web servers can be used, CDN postulating the Web such as OpenWeb [25] would be proper. Alternatively, CDN postulating P2P [26] would be proper if web servers cannot be used or are not allowed to be used for some reason.

In the future, if the next generation network (NGN) has been well-wired, it may bear a chance of transmitting vast amounts of information directly.

5.5 Using Chord

To our knowledge, there is no versatile approach to enable information dissemination and reputation aggregation simultaneously in distributed systems. If information dissemination is simply required, application layer multicast or flooding would be the most common and efficient means. For that purpose, Pastry [28] which has a mesh topology and/or Kademlia [29] which has a tree topology must be appropriate. Additionally, in some structured overlay networks such as Skip Graph [18] and Chord# [27], range query is available for information dissemination; efficient range query for aggregation is also proposed as mentioned in Section 2.2. However, when and who should invoke aggregation query could be another difficult issue.

The most important reason why we have employed Chord is because of its simple ring topology. Using Chord, a pathway of a circular board is directly corresponding to its topology. This means that information is disseminated to consumer peers and reputations are collected at producer peers just by forwarding circular boards along the topology once. If non-ring topologies are used, at least pathway control mechanisms are absolutely necessary and it would be a thorny issue to collect distributed circular boards in a P2P manner. Again, we believe that the most significant part of our work is to enable information dissemination and reputation aggregation simultaneously. Performing information dissemination and reputation aggregation in different phases is no more than a combination of existing works introduced in Section 2.1 and 2.2 or the well-known approaches of overlay networks described above.

6. Conclusion

In this paper, we have proposed to apply the circular board method based on Chord to user centric media to disseminate information and aggregate reputation efficiently in a P2P manner. The proposed system makes it possible to effectively collect the reputation from users at the same time when each piece of UGC passes through user terminals. The results of simulations have provided insights about trade-offs between network resource consumed and time required for information dissemination and reputation aggregation. As future work, it is planned to customize the original topology of Chord and explore other topology such as layered P2P networks [30] to make the efficiency higher.

Acknowledgments This work was supported by JSPS KAKENHI Grant Number 23650033.

References

[1] Cisco Systems: Entering the Zettabyte Era, White paper, Cisco Systems (2011).

[2] Sakaki, T., Toriumi, F. and Matsuo, Y.: Tweet trend analysis in an emergency situation, *Proc. Special Workshop on Internet and Disasters, SWID '11*, pp.3:1–3:8, ACM (online), DOI: 10.1145/2079360.2079363 (2011).

[3] Parameswaran, M., Susarla, A. and Whinston, A.: P2P networking: An information sharing alternative, *Computer*, Vol.34, No.7, pp.31–38 (online), DOI: 10.1109/2.933501 (2001).

[4] Montresor, A. and Abeni, L.: Cloudy Weather for P2P, with a Chance of Gossip, *2011 IEEE International Conference on Peer-to-Peer Computing* (*P2P*), pp.250–259, IEEE (2011).

[5] Stoica, I., Morris, R., Karger, D., Kaashoek, M.F. and Balakrishnan, H.: Chord: A scalable peer-to-peer lookup service for Internet applications, *Proc. 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '01*, pp.149–160, ACM (online), DOI: http://doi.acm.org/10.1145/383059.383071 (2001).

[6] Wu, S., Hofman, J.M., Mason, W.A. and Watts, D.J.: Who says what to whom on Twitter, *Proc. 20th International Conference on World Wide Web, WWW '11*, pp.705–714, ACM (2011).

[7] Costa, P. and Frey, D.: Publish-subscribe tree maintenance over a DHT, *25th IEEE International Conference on Distributed Computing Systems Workshops, 2005*, pp.414–420, IEEE (2005).

[8] Zhu, Y. and Hu, Y.: Ferry: A P2P-based architecture for content-based publish/subscribe services, *IEEE Trans. Parallel and Distributed Systems*, Vol.18, No.5, pp.672–685 (2007).

[9] Rao, W., Vitenberg, R. and Tarkoma, S.: Towards optimal keyword-based content dissemination in DHT-based P2P networks, *2011 IEEE International Conference on Peer-to-Peer Computing*, pp.102–111, IEEE (2011).

[10] Mega, G., Montresor, A. and Picco, G.: Efficient dissemination in decentralized social networks, *2011 IEEE International Conference on Peer-to-Peer Computing*, pp.338–347, IEEE (2011).

[11] Kermarrec, A.-M., Pace, A., Quéma, V. and Schiavoni, V.: NAT-resilient gossip peer sampling, *29th IEEE International Conference on Distributed Computing Systems, ICDCS'09*, pp.360–367, IEEE (2009).

[12] Voulgaris, S., Gavidia, D. and Van Steen, M.: Cyclon: Inexpensive membership management for unstructured p2p overlays, *Journal of Network and Systems Management*, Vol.13, No.2, pp.197–217 (2005).

[13] Magharei, N., Rejaie, R. and Guo, Y.: Mesh or multiple-tree: A comparative study of live p2p streaming approaches, *26th IEEE International Conference on Computer Communications, INFOCOM 2007*, pp.1424–1432, IEEE (2007).

[14] Wang, F., Liu, J. and Xiong, Y.: Stable peers: Existence, importance, and application in peer-to-peer live video streaming, *The 27th Conference on Computer Communications, INFOCOM 2008*, pp.1364–1372, IEEE (2008).

[15] Zhou, R., Hwang, K. and Cai, M.: Gossiptrust for fast reputation aggregation in peer-to-peer networks, *IEEE Trans. Knowledge and Data Engineering*, Vol.20, No.9, pp.1282–1295 (2008).

[16] Yajima, T., Matsumoto, A. and Shigeno, H.: PTrust: Provisional Value based trust for reputation aggregation in peer-to-peer networks, *2011 1st International Symposium on Access Spaces* (*ISAS*), pp.180–185, IEEE (2011).

[17] Abe, K., Abe, T., Ueda, T., Ishibashi, H. and Matsuura, T.: Aggregation Skip Graph: A Skip Graph Extension for Efficient Aggregation Query over P2P Networks, *International Journal On Advances in Internet Technology*, Vol.4, No.3 and 4, pp.103–110 (2012).

[18] Aspnes, J. and Shah, G.: Skip graphs, *Proc. 14th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '03*, Philadelphia, PA, USA, Society for Industrial and Applied Mathematics, pp.384–393 (online), available from ⟨http://dl.acm.org/citation.cfm?id=644108.644170⟩ (2003).

[19] Nock, R. and Nielsen, F.: On weighting clustering, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.28, No.8, pp.1223–1235 (2006).

[20] Agoop Corp.: Data analysis for smartphone packet communication (online), available from ⟨http://www.agoop.co.jp/solutions/reports/report_3/report.html⟩ (accessed 2013-03-24) (in Japanese).

[21] Kwak, H., Lee, C., Park, H. and Moon, S.: What is Twitter, a social network or a news media?, *Proc. 19th International Conference on World Wide Web, WWW '10*, pp.591–600, ACM (2010).

[22] Radio, N., Zhang, Y., Tatipamula, M. and Madisetti, V.: Next-Generation Applications on Cellular Networks: Trends, Challenges, and Solutions, *Proc. IEEE*, Vol.100, No.4, pp.841–854 (online), DOI: 10.1109/JPROC.2011.2182092 (2012).

[23] Browserscope: A community-driven project for profiling web browsers (online), available from ⟨http://www.browserscope.org/⟩ (accessed 2013-07-10).

[24] Nygren, E., Sitaraman, R. and Sun, J.: The Akamai Network: A Platform for High-Performance Internet Applications, *ACM SIGOPS Operating Systems Review*, Vol.44, No.3, pp.2–19 (2010).

[25] Sakurauchi, Y., McGeer, R. and Takada, H.: OpenWeb: Seamless Proxy Interconnection at the Switching Layer, *International Journal of Networking and Computing*, Vol.1, No.2 (online), available from ⟨http://www.ijnc.org/index.php/ijnc/article/view/22⟩ (2011).

[26] El Dick, M., Pacitti, E. and Kemme, B.: A highly robust P2P-CDN under large-scale and dynamic participation, *First International Conference on Advances in P2P Systems, AP2PS'09*, pp.180–185, IEEE (2009).

[27] Schütt, T., Schintke, F. and Reinefeld, A.: Range queries on structured overlay networks, *Comput. Commun.*, Vol.31, No.2, pp.280–291 (online), DOI: 10.1016/j.comcom.2007.08.027 (2008).

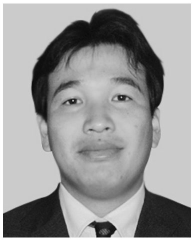
[28] Rowstron, A.I.T. and Druschel, P.: Pastry: Scalable, Decentralized

Object Location, and Routing for Large-Scale Peer-to-Peer Systems, *Proc. IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg, Middleware '01*, pp.329–350, Springer-Verlag (online), available from ⟨http://dl.acm.org/citation.cfm?id=646591.697650⟩ (2001).

[29] Maymounkov, P. and Mazieres, D.: Kademlia: A peer-to-peer information system based on the xor metric, *Peer-to-Peer Systems*, pp.53–65, Springer (2002).

[30] Garces-Erice, L., Biersack, E.W., Felber, P.A., Ross, K.W. and Urvoy-Keller, G.: Hierarchical peer-to-peer systems, *Euro-Par 2003 Parallel Processing*, pp.1230–1239 (2003).

Yoshio Sakurauchi received his B.E. and M.E. degrees from Ritsumeikan University in 2009 and 2011, respectively. He is currently a Ph.D. candidate at Graduate School of Science and Engineering, Ritsumeikan University. His research interests are distributed systems, user centric media and network services.



Hideyuki Takada received his B.E., M.E. degrees in information science from Kyoto University in 1991 and 1993 respectively. He also received his Ph.D. degree in social informatics from Kyoto University in 2001 while he worked for Mitsubishi Electric Corporation from 1993 to 2003. After he worked as a researcher at Kyoto University from 2004 to 2006, he is currently a professor at College of Information Science and Engineering, Ritsumeikan University. His research interests include distributed systems, collaboration support systems and learning support systems. He is a member of IEICE, JSET, IEEE Computer Society and ACM.