

P2P File Sharing Networks Allowing Participants to Freely Assign Structured Meta-data to Files

Kei Ohnishi
Network Design Research
Center, Kyushu Institute of
Technology
3-8-1 Asano, Kokura-Kita-ku,
Kitakyusyu, Fukuoka
802-0001, JAPAN
ohnishi@ndrc.kyutech.ac.jp

Kaori Yoshida
Kyushu Institute of Technology
680-4 Kawazu, Iizuka,
Fukuoka 820-8502, JAPAN
kaori@ai.kyutech.ac.jp

Yuji Oie
Kyushu Institute of Technology
680-4 Kawazu, Iizuka,
Fukuoka 820-8502, JAPAN
oie@cse.kyutech.ac.jp

ABSTRACT

The present paper proposes the concept of peer-to-peer (P2P) file sharing networks that allow participants (peers) to freely assign structured meta-data to files. As a concrete example, we consider an unstructured P2P network using vectorized Kansei (human sensitivity) information as structured meta-data for file search. Vectorized Kansei information as meta-data indicates what participants feel to their own files and is assigned by the participant to each of their own files. Therefore, vectorized Kansei information is a sort of structured meta-data that people can freely assign to files. A search query also has the same form of vectorized Kansei information and indicates what participants want to feel to files that they will eventually obtain. A method that enables file search using vectorized Kansei information is the *Kansei query-forwarding method*, which probabilistically propagates a search query from a peer making the query to peers that are likely to hold more files having meta-data that is similar to the query. The similarity between the search query and the meta-data is measured in terms of their dot product. From the viewpoint of P2P file sharing, in which all of the peers are equal in terms of function, it is not good for certain peers to have an advantage in P2P file search due to their Kansei information. Therefore, the simulation experiments herein examine if the Kansei query-forwarding method can provide equal search performance for all peers in a network in which the Kansei information of the peers and the tendency of the peers with respect to file collection are diverse. The simulation results show that the Kansei query forwarding method and a random-walk-based query forwarding method, for comparison, work effectively in different situations and are complementary.

Categories and Subject Descriptors

H.3.4 [Information Storage and Retrieval]: Systems

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Infoscale 2007 June 6-8, 2007, Suzhou, China
Copyright 200X ACM 978-1-59593-757-5 ...\$5.00.

and Software—*distributed systems, information networks*

General Terms

Algorithms, Design, Human Factors

Keywords

P2P file sharing, meta-data, query forwarding, human

1. INTRODUCTION

Recently, peer-to-peer (P2P) network models have attracted a great deal of attention. The concept of the P2P network model is completely different from that of a conventional client-server network model. While a conventional server-client network model explicitly distinguishes hosts providing services (servers) from hosts receiving services (clients), a P2P network model does not assign fixed roles to hosts. Hosts composing P2P networks, referred to as *peers*, can be both server and client, so that P2P networks can manage services in a decentralized manner.

One of the applications of P2P networks that people are interested in is a distributed storage system for file sharing. A distributed storage system for file sharing provides a large amount of storage by accumulating unused storage of hosts, enabling large amounts of data to be stored and shared without the need for a costly file server. According to [4], there are several forms in P2P networks for file sharing. We can roughly classify P2P networks for file sharing into two types. One type is P2P networks that have a mechanism to manage file locations in a network. The other type is P2P networks that do not have such a mechanism.

One of the mechanisms to manage file locations in P2P networks is a distributed hash table (referred to hereinafter as DHT) [6]. P2P networks using DHT instruct peers in a network as to which files they require and how peers are connected to one another. Therefore, these networks are generally referred to as structured P2P networks. An exhaustive survey on structured P2P networks is provided in [4]. In addition, P2P networks that do not have a mechanism to manage file locations in a network, which are referred to as unstructured P2P networks, cannot provide peers with the locations of requested files. Therefore, a query-forwarding method is needed to find requested files. Most studies on unstructured P2P networks have examined query-forwarding methods [5][1].

With respect to information search techniques, the simplest method is to search for objects by explicitly expressing target objects such as file names. However, in order to express the target objects precisely, we need to know what to search for in advance, and such a situation should be rare. Therefore, searching for objects from pieces of information included in target objects has been considered. The present Web search techniques adopt such a strategy. Pieces of information included in target objects are not target objects in and of themselves, but can be thought of as alternative forms pointing to target objects, and the alternative forms can be regarded as a sort of meta-data of target objects. Furthermore, as a variation of pieces of information included in target objects, we can use features that are extracted by processing the target objects.

Proceeding from using pieces of information included in search objects or features obtained from search objects as meta-data, we consider a file search based on meta-data that participants freely assign to files. One such type of meta-data is *Kansei* information, which in Japanese means ‘human sensitivity’ information. There have been several attempts to use *Kansei* (human sensitivity) information in a manner such that how people feel to objects is assigned as the meta-data of the participant. Such information search techniques are referred to as *Kansei* information retrieval techniques, and there exist several concrete *Kansei* information retrieval methods [9][7].

In the present paper, we propose the concept of file sharing P2P networks that allow the participants (peers) to freely assign structured meta-data to files. As a concrete example of this concept, we consider an unstructured P2P network using vectorized *Kansei* (human sensitivity) information as structured meta-data for file search. The vectorized *Kansei* information as meta-data indicates what people feel to their own files, and is assigned by the participant to each of their own files. The vectorized *Kansei* information is, therefore, a sort of structured meta-data that people can freely assign to files. A search query also has the same form of the vectorized *Kansei* information and indicates what people want to feel to files that they will eventually obtain. A mechanism that enables file search using the vectorized *Kansei* information is the *Kansei query-forwarding method*, which probabilistically propagates a search query from a peer making a query to peers that are likely to hold more files whose meta-data is similar to the query. The similarity between the search query and the meta-data is measured by their dot product.

The present paper is organized as follows. Section 2 briefly describes related studies. We describe the motivation behind the present work and the basic concept of the proposed network in Section 3. In Section 4, we describe the proposed P2P file sharing network that uses *Kansei* information as meta-data, as well as the *Kansei* query-forwarding method. The usefulness of the proposed network is evaluated in terms of mainly the *Kansei query-forwarding method* in Section 5. Finally, Section 6 presents conclusions and areas for future study.

2. RELATED STUDIES

In this section, we briefly describe studies related to *Kansei* information retrieval systems as well as P2P networks with query-forwarding methods using meta-data.

According to [3], in which a variety of P2P networks with query-forwarding methods using meta-data are introduced,

there are two types of P2P networks that use meta-data. One type of P2P network can provide deterministic routing for queries with a form of meta-data by means of a mechanism to locate files without question to which meta-data is assigned, such as DHT. The other type of P2P network employs non-deterministic routing due to the lack of a mechanism to locate files without question. Most of the P2P networks introduced in [3] use text data as meta-data.

The P2P network presented herein does not have a mechanism to manage file locations in the network. The reason for choosing this form of P2P network is as follows. Since *Kansei* information should vary among individuals, a map between files and *Kansei* information as meta-data for the files could be a one-to-many map, it is difficult to generate a framework of P2P networks that is equipped with deterministic routing, such as DHT-based P2P networks, in which the basic assumption is that a map between files and meta-data for the files is a many-to-one map, in the present study. The P2P network proposed herein uses individual *Kansei* information (impression) with a form of a numerical vector having a fixed length as meta-data for the files. Such P2P networks have not yet been investigated.

File search in the proposed P2P network is conducted using a query forwarding method that propagates a query in the form of vectorized *Kansei* information over a network, which is referred to herein as the *Kansei* query forwarding method. The *Kansei* query forwarding method basically forwards a query to peers that have more files with meta-data similar to the query. For the case in which a map between files and their meta-data is a many-to-one map, such a query forwarding method is referred to as semantic routing [2][8][3], although the focus of semantic routing is literally semantics and semantic routing usually employs text data extracted from text files as their meta-data.

Next, existing *Kansei* information retrieval systems [9][7], which are built on server-client networks, have a common approach to information retrieval. These methods are composed of two spaces and a map between them. One of the two spaces is of search objects, such as multi-media content. More precisely, one of the two spaces is of features extracted from the objects. The other space is of the impressions that participants form regarding the search objects. The fundamental difference among existing systems is in a way to build a map between the two spaces mentioned above. On the other hand, the P2P network presented in this paper does not retrieve information by means of such a map between the two spaces.

3. MOTIVATION AND CONCEPT

In the previous sections, we described the background and the objectives of this study from the viewpoint of information as objects of search and clues for the search. Although the goal of information retrieval is usually to find desired information, the motivation of the present study is not to develop more efficient search techniques. Rather, we focus herein on information as intermediates that facilitate communication between humans. Although conventional information retrieval techniques recognize the identity of information itself, they do not assume the identity of the participant holding the information. Since there may be personal reasons for holding specific information, if we can understand these reasons during information retrieval, we may improve inter-participant communications beyond simple in-

formation retrieval.

The proposed approach, considering the identities of information holders in information retrieval systems, allows participants to assign meta-data to their own files as they see fit. In this case, different meta-data can be assigned to identical information by different participants. This situation is not desirable in terms of efficient information retrieval. However, once a participant encounters other participants that assign the same or similar meta-data to identical information through information retrieval, it is expected that the participants will share a great deal of information.

Next, we have to consider the form of meta-data that participants are allowed to assign to their own information in order to facilitate communication between humans, as mentioned above. An extreme method is to allow participants to assign meta-data in a completely free manner. However, this would not realize practical information retrieval because the size of the search space can become large. Therefore, in the present study, we consider structured meta-data, such as vectorized information, and the number of structured meta-data types that participants can assign to their own information is finite.

Finally, we must examine whether meta-data that a certain search technique uses for information retrieval is useful for improving communication between humans, in which case the first information retrieval process is simply a trigger. However, this is difficult to examine because a model of human communications beyond actions of information retrieval is needed. Therefore, the simulation experiments in Section 5 examine whether such meta-data helps to provide opportunities for a participant to meet other participants with a similar sense regarding their files.

4. KANSEI INFORMATION IN P2P NETWORKS

4.1 Vectorized Kansei Information as Meta-data

The simplest conventional file search mechanism requires explicit information to be given regarding search objects, such as file names or pieces of file names. Here, we consider file search with meta-data as a query, such that file search begins by specifying a name or feature of a set to which search objects belong and eventually reaches concrete desired objects. A name or feature of a set to which files belong is upper concept than a file itself.

A set to which files belong consists of two sub-sets. One sub-set is related to a category, which is objective, such as the “music” category. The other sub-set is related to the impression or feeling of participants regarding the contents of files, which is quite subjective. Some participants may consider certain music to be pleasant, while other participants consider the same music to be unpleasant. In this case, a “music” file may belong to different subsets of impression or feeling, depending on the participant.

Next, one method of representing which sets a file belongs to is to use words such as “music” and “pleasant”. Here, we use not words, but vectors of numerical values of fixed length, which enables the calculation of the dot product mentioned later herein. Owners of files are not allowed to completely freely assign vectors of numerical values as meta-data to their own files. Vectors of numerical values

that owners of files are allowed to assign to their own files are determined in advance, and participants can assign a vector of numerical value to each file. For example, a file that belongs to the “music” category is represented by $(\text{text, music, movie}) = (0, 1, 0)$, where “1” indicates that the file belongs to the category represented by the category name, and “0” indicates that the file does not belong to the category. Meanwhile, with respect to a numerical vector representing impression or feeling, a file is, for example, represented by $(\text{light, pleasant, beautiful}) = (-1, 0, 1)$, where “1” indicates that the content of the file gives the impression represented by the impression word, “0” indicates that the content of the file gives the impression represented by the impression word to an extent, and “-1” indicates that the content of the file does not give the impression represented by the impression word at all.

Participants may wish to add elements of a numerical vector to represent files during file sharing. For example, “complicated” is added into a numerical vector representing impression or feeling as its element. In a situation in which the elements of a numerical vector representing a file increase dynamically, the participants need to guarantee that consistent numerical vectors are assigned to their own files. One solution for this is to introduce a server that manages elements of a numerical vector to represent files. The server accepts the registration of elements from participants and then forms a numerical vector by arranging the elements from the left side in old order. Participants obtain information on a new numerical vector regularly. In this way, even if a participant does not have information on a new numerical vector, there is no inconsistency in terms of the order of elements between a numerical vector that the participant knows and a new numerical vector up to the length of the numerical vector that the participant knows.

In addition, it may not be practical to force participants to assign a numerical vector to all of their own files with the increasing number of files. One solution for this is to assign a certain numerical vector, for example $(\text{text, music, movie, light, pleasant, beautiful}) = (0, 0, 0, 0, 0, 0)$, to all files as a default or to simply ignore the elements of numerical vectors that are not filled in by participants when performing operations on numerical vectors, as explained later herein.

While each file is represented by a numerical vector as mentioned above, a peer that would hold several files is represented by the sum of all numerical vectors for the files. The representation of a peer is used for query forwarding mentioned later. For example, when a peer holds three files with vectors as meta-data, the representation of the peer is the sum of the three vectors. In addition, it is possible to make participants select files for the representation of peers.

A search query made by a peer is represented in the form of a numerical vector assigned to a file as meta-data, such as $(\text{text, music, movie, light, pleasant, beautiful}) = (1, 0, -1, 1, 1, 0)$. A search query is, as explained in the following section, propagated with high probability to peers in which the representations give larger values of the dot product with the query. The number of hops that is allowed for a query is limited to N_h , and lists of files of peers to which the query was propagated are given to a peer making the query. The peer making the query can select and download some of the files in these lists, if the peer desires so.

4.2 Kansei Query Forwarding

The proposed P2P network for file sharing has a mechanism that varies the probability with which a query is propagated on network links through file searches. The probability assigned to network links is not managed by particular hosts, but varies in a self-organizing manner through a file search performed by each peer. In the following, we will explain the Kansei forwarding method, which plays a central role in self-organization as well as in file search.

Here, we let the number of walkers propagating a query be one, although more than one walker can be used. The walker considers a numerical vector \mathbf{q} , such as $(1, 0, -1, 1, 1, 0)$, to be a query. The walker obtains values of the dot product between \mathbf{q} and N_n numerical vectors representing the peers adjacent to the peer at which the walker is currently located, \mathbf{r}_i ($i = 1, 2, \dots, N_n$). The probability with which the k -th peer in the N_n peers will be selected as the peer that the walker will hop, ps_k , is denoted by Equation (1).

$$ps_k = \frac{dp_k - \min_{dp} + 1}{\sum_{i=1}^{N_n} \{dp_i - \min_{dp} + 1\}}, \quad (1)$$

where $dp_k = \mathbf{q} \cdot \mathbf{r}_k$, and \min_{dp} is the smallest value among the N_n dot products. Since the dot product can be negative, all of the dot products are forced to be positive by means of the term $-\min_{dp} + 1$. In doing so, the peer to which the walker will hop next is determined with a probability proportional to the dot product between its numerical vector and the query. However, if the peers that a walker has visited once in the present file search are again selected, according to Equation (1), the walker selects a peer that has not been visited by the walker for its next hop. An example in which the proposed Kansei query-forwarding method decides a peer as the next hop of a walker is shown in Figure 1.

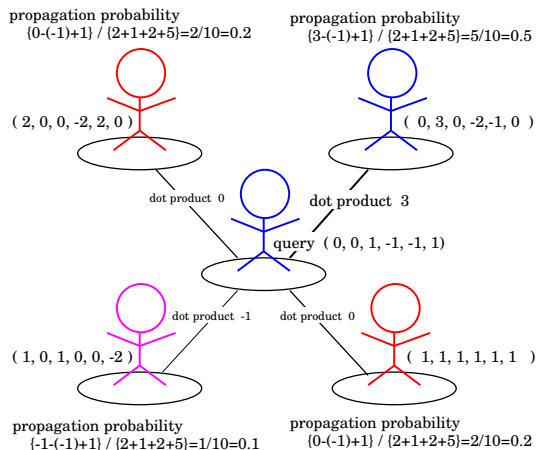


Figure 1: Example of query forwarding. Circles represent peers, and a query has reached the peer in the center. The representations of the peers linked to the peer in the center are described near the circles. Calculating the values of the dot product between the query using a form of a numerical vector and the numerical vectors representing the peers, the probability of the query is propagated to each peer, as given by Equation (1).

Lists of files of peers that a walker visited within a limited

number of hops, N_h , are provided for a peer making a query. Then, if the peer making the query finds files that it wants in the lists, the peer can download files within a limited number of downloads, as mentioned in the next section. In addition, the peer making the query also obtains the numerical vectors representing the files that it downloaded. When a peer making a query downloads several files from peers that a walker visited in the present file search, files held by the peer making the query vary. Consequently, the representation of the peer, which is the sum of all of the numerical vectors attached to its own files, also varies. The peer making the query can also modify the numerical vectors of the files that it has downloaded. Therefore, the probability with which a query is propagated on links to the peer making the query varies.

4.3 P2P File Sharing

The proposed P2P network has a restriction on the number of downloads to peers in the network, such that peers are not allowed to download an unlimited number of files from other peers, but rather are allowed to download files according to the number of times that other peers have downloaded their files. When a peer has had one of its own files downloaded by a peer, the peer increases by one the number of files that it can download from other peers. On the other hand, when a peer has downloaded one file from another peer, it decreases by one the number of files that it can download from other peers. Peers that hold popular files are likely to increase the number of times that they can download files from others. However, every peer is given a fixed number of times that it can download files, N_D , when it participates in the network, because without doing so, the P2P network does not work as a system for file sharing. After performing N_D complete downloads, peers can only increase the number of downloads by having their own files downloaded by other peers.

As explained in the previous section, numerical vectors representing both query and file are given by peers, and changes in the numerical vectors cause changes both in the probability with which a query is propagated to the peers and in the search results. Therefore, acts of giving numerical vectors to files are strategic factors for the purpose of file sharing. However, even if queries have frequently reached certain peers, the peers can not increase the number of downloads unless their own files are actually downloaded by other peers.

5. EXPERIMENTAL EVALUATION

5.1 Objective

The P2P file search considered in the present paper relies on Kansei information of participants in the network. Kansei information is generally different for each participant, so that the search performance for each participant might not be the same. However, from the viewpoint of P2P file sharing, in which all of the participants (peers) are equal in terms of function, it is not good for certain participants to take advantage of P2P file search due to their Kansei information. Therefore, we experimentally examine whether the Kansei query-forwarding method can provide equal search performance for the situation in which the Kansei information of participants in the network is diverse. Therefore, we need to model differences among peers in Kansei information, as

well as differences among peers with respect query tendencies. We will describe the modeling of these differences later herein.

5.2 P2P Simulation Model

With respect to network structure, the number of peers, n_p , present in the network is 200. The network topology used is full-mesh. The reason for choosing a full-mesh topology is because in the present paper we intend to have a network topology that is stabilized through repeating file searches, not by having peers choose their own links. However, a full-mesh topology might be impractical with an increasing number of peers, and, in practice, it may be necessary to have all peers select a fixed number of links. Next, the number of files types distributed over the network, f_k , is the same as the number of peers, that is 200. Initially, each peer has a different file.

5.3 Model of Kansei Information

First, the numerical vectors given to files have only one element, and this element takes the value of either “-1” or “+1” and does not take the value of 0. The Kansei information of each peer (participant) in the network is represented as the value (-1 or +1) that each of the n_p peers gives to the f_k types of files that are distributed over the network.

Although peers cannot assign impressions or feelings to files prior to seeing the files, we define which values each peer assigns to f_k types of files in advance. Therefore, the Kansei information of each peer for f_k types of files is represented as a numerical vector with f_k elements, as $(-1, -1, \dots, +1)$, in which the i -th element corresponds to the i -th file type among f_k file types. We hereinafter refer to the numerical vector as the *Kansei vector*.

The Kansei vector of each peer is generated by modifying a given numerical vector with f_k elements. The given numerical vector without modification is referred to hereinafter as the *prototypical Kansei vector*. The prototypical Kansei vector is modified by flipping a value of each element (-1 to +1 or +1 to -1) with probability p_b . The expected Hamming distance between two Kansei vectors is generated according to following equation.

$$f_k \times \{p_b \times (1 - p_b) + (1 - p_b) \times p_b\} = 2f_k \cdot p_b(1 - p_b),$$

where f_k is the number of file types distributed over the network.

Concretely, we use the three types of prototypical Kansei vectors below to produce the Kansei vector of each peer. The three prototypical Kansei vectors differ from each other only in the ratio of “-1”.

Prototypical Kansei vector (1): The 1st through 100th elements in the vector take the value of -1, and the other elements take the value of +1.

Prototypical Kansei vector (2): The 1st through 120th elements in the vector take the value of -1, and the other elements take the value of +1.

Prototypical Kansei vector (3): The 1st through 160th elements in the vector take the value of -1, and the other elements take the value of +1.

Although we cannot show the observed result due to the space limitations, the average number of “-1” values becomes approximately 100 at $p_b = 0.5$ independent of which prototypical Kansei vector is used.

5.4 Model of Tendency on Queries

A search query is of the same form as Kansei information as meta-data, and as a value of either “-1” or “+1”. The tendency of a peer with respect to queries is expressed by the probability with which “-1” is chosen as a query by the peer. This probability is denoted by $p \in [0, 1]$. We use two methods to determine the value of p for each peer in the network below.

Random determination of p

This method literally determines p of each peer by a uniform random number in $[0, 1]$.

Kansei-correlated determination of p

This method determines p of each peer using the following equation:

$$p = \frac{N_{one}}{f_k},$$

where N_{one} is the number of -1 values in the Kansei vector of the peer, f_k is the number of file types distributed over the network (length of Kansei vector), and f_k is 200.

5.5 File Search and Sharing

The number of walkers that propagate a query is just one. The number of hops that a walker is allowed during one file search is three, that is, E_h is 3. Each peer is initially given permission to download five files, that is, N_D is 5. For example, when a peer makes a query of “-1”, and then some of the peers that a walker visited have files belonging to the set of “-1” that the peer making the query does not have, the peer chooses one of these files randomly and downloads the file. If the walker cannot find a file that the peer does not have, the file search is finished.

5.6 Evaluation Criteria

Query forwarding methods are evaluated according to the number of file searches that are required so that all of the peers will hold more than $2n_p/5$ files, where n_p is the total number of file types distributed over the network and n_p is 200. This number of file searches is referred to as the *convergence time* and the state in which all of the peers hold $2n_p/5$ or more files is defined as the *convergence*. A failure of convergence indicates that convergence does not occur within the number of file searches given in advance, E_s . We set the value of E_s to 30,000. Quick convergence means that all of the peers in the network obtain similar search performance.

We count the following three cases as one file search: (1) a file that a peer making a query desires was found within three hops ($N_h = 3$) of a walker, (2) a file that a peer making a query desires was not found within three hops of a walker, and (3) a peer that did not have permission to download files from others when its turn arose. We do not count as one file search the situation in which some peer made a query but previously held all files belonging to the set represented by the query.

5.7 Simulation Settings

As mentioned earlier, three types of prototypical Kansei vectors are used, prototypical Kansei vectors (1), (2), and (3). These vectors are modified into the Kansei vectors of peers by flipping the values of each of their elements with probability p_b . In addition, two methods by which to determine the p values of peers are used. We will test all six combinations of three types of prototypical Kansei vectors

and the two methods by which to determine the p values of peers. For each combination with a value of p_b less than 0.5, we will observe the average convergence time over independent runs that resulted in convergence and the number of convergence failures during 500 independent runs.

For comparison, as a query forwarding method, we employ the *one-walker random walk*. The one-walker random walk allows a walker that knows a search query to randomly decide the next peer to visit. However, in one file search, a walker does not revisit the peers that it has already visited.

5.8 Results and Discussion

The simulation results are shown in Figures 2 and 3. Figure 2 shows the average convergence time and the number of convergence failures for the Kansei query-forwarding method and the one-walker random walk in the case of random determination of p , and Figure 3 shows the average convergence time and the number of convergence failures for the Kansei query-forwarding method and the one-walker random walk in the case of Kansei-correlated determination of p .

The obtained results do not show the success rate in one file search by the used query forwarding methods, but rather indicate the success rate of the convergence defined in Section 5.6. The success convergence could result from smooth circulation of the permission to perform file downloads, which initially is given to each peer. Meanwhile, failure to converge may be a result of the permission to download being biased to specific peers or may be a result of the query forwarding method used being unable to provide reliable search performance for most peers.

According to Figure 2 (comparing Figures 2(a) and 2(c) for Kansei query forwarding and Figures 2(b) and 2(d) for one-walker random walk), when using the random determination of p , the Kansei query forwarding method is superior to or approximately equal to the one-walker random walk in terms of the ability to induce quick convergence, that is, in providing similar search performance for all of the peers. Specifically, when using small values of p_b , the Kansei query forwarding method is superior to the one-walker random walk, and the ability of these two methods becomes more similar with increasing values of p_b .

When the difference in the ability to induce convergence between the Kansei query forwarding method and the one-walker random walk was the largest, that is, when using the prototypical Kansei vector (1) and $p_b = 0$, vectorized Kansei information as meta-data works efficiently as search tags, and the randomness in file search provides redundancy. In addition, since the number of “-1” and “+1” values in the Kansei vectors of the peers are the same and the p values of the peers are randomly determined, the total number of files corresponding to “-1” and that corresponding to “+1” in the network are approximately the same at any time during file search. Therefore, all of the peers with their own p values had approximately equal abilities to obtain and provide files, independent of p value. When using Kansei vector (1) and the larger values of p_b , vectorized Kansei information as meta-data no longer works as efficient search tags, and randomness is needed in the file search. However, unreliable search tags cause file search by the Kansei query forwarding method to be similar to random search. As a result, with increasing values of p_b , the abilities of the Kansei query forwarding method and the one-walker random walk to induce

convergence become increasingly similar.

Next, when using prototypical Kansei vector (2), as in the case of using prototypical Kansei vector (1), vectorized Kansei information as search tags is still reliable if the value of p_b is small. Therefore, the ability of the Kansei query forwarding method to induce convergence is better than that of the one-walker random walk. However, the number of “-1” values in the Kansei vectors of most peers should be larger than the number of “+1” values. Since the p values of the peers were randomly determined in this situation, peers that have a high probability to produce a query of “-1” could easily obtain and frequently provide files corresponding to “-1”. The total number of which in the network should be larger than the number of files corresponding to “+1” at any time during the file search. Meanwhile, peers that have high probability to produce a query of “+1” could not easily obtain and frequently provide files corresponding to “+1”. This should result in a difference in search performance between the peers, and consequently, the ability of the Kansei query forwarding method to induce convergence in the case of using prototypical Kansei vector (2) and small values of p_b would be worse than that in the case of using prototypical Kansei vector (1) and small values of p_b . Furthermore, in the case of using prototypical Kansei vector (3) and small values of p_b , it should be more difficult for peers that have a high probability to produce a query of “+1” to easily obtain and frequently provide files corresponding to “+1”, compared to the case of using prototypical Kansei vector (2) and small values of p_b . However, even when prototypical Kansei vector (2) or (3) is used, if the value of p_b becomes larger, the numbers of “-1” and “+1” values in the Kansei vectors of the peers become more similar and randomness is needed in the file search. Therefore, the abilities of the Kansei query forwarding method and the one-walker random walk to induce convergence would become closer with the increasing value of p_b .

On the other hand, according to Figure 3 (comparing Figures 3(a) and 3(c) for Kansei query forwarding and Figures 3(b) and 3(d) for the one-walker random walk), when using the Kansei-correlated determination of p , the one-walker random walk is superior to or approximately equivalent to the Kansei query forwarding method in terms of the ability to provide similar search performance for all of the peers. Specifically, when using small values of p_b , the one-walker random walk is superior to the Kansei query forwarding method, and the ability of these two methods becomes closer with increasing values of p_b .

When the difference in the ability to induce convergence between the Kansei query forwarding method and the one-walker random walk was the largest, that is, when using prototypical Kansei vector (3) and $p_b = 0$, every peer has the same tendency during file collection, i.e., the same value of p . In this case, even if a peer making a query randomly chooses other peers, it is likely that the chosen peers have files that the peer making the query desires. Therefore, the one-walker random walk yields better performance with respect to inducing convergence. In this situation, vectorized Kansei information as meta-data works efficiently as search tags. Since the number of “-1” values in the Kansei vectors of the peers is larger than the number of “+1” values, and the p value of the peers is proportional to the number of “-1” values, all of the peers need to have approximately the same number of files corresponding to “-1” at any time during the

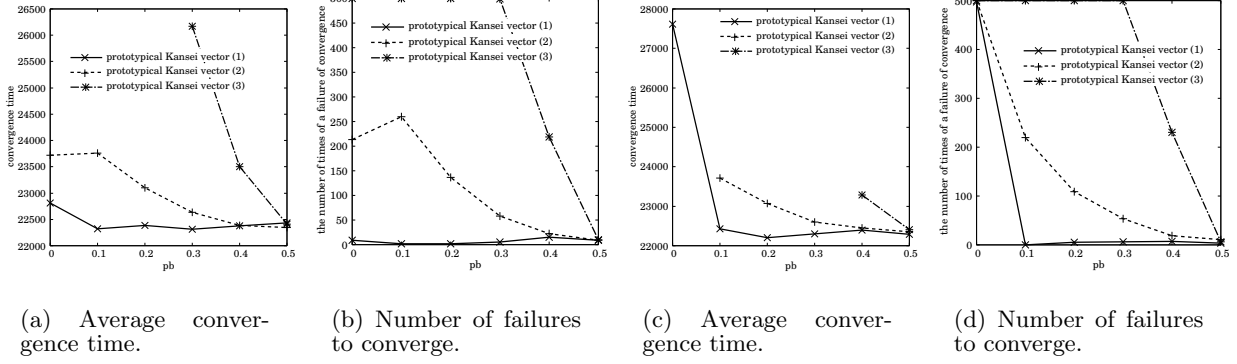


Figure 2: Simulation results obtained using random determination of p . Sub-figures (a) and (b) are for the Kansei query-forwarding method, and sub-figures (c) and (d) are for a one-walker random walk.

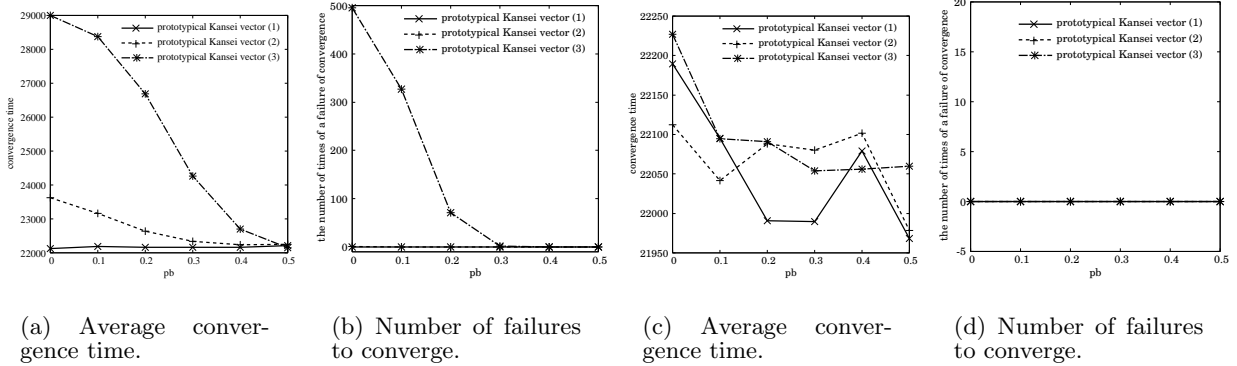


Figure 3: Simulation results obtained using the Kansei-correlated determination of p . Sub-figures (a) and (b) are for the Kansei query-forwarding method, and sub-figures (c) and (d) are for a one-walker random walk.

file search in order to achieve quick convergence. However, specific peers are given more opportunities for their files to be downloaded by the Kansei query forwarding method, although we cannot show the observed result due to the space limitations. The reason for this is because these specific peers happened to increase the number of files corresponding to “-1”. Consequently, this causes positive feedback to be yielded such that the initial small advantage in the number of files corresponding to “-1” over other peers grows larger and larger with time.

Next, when using prototypical Kansei vector (2), as in the case of using prototypical Kansei vector (3), the Kansei query forwarding method produces the above-mentioned positive feedback to specific peers if the value of p_b is small, because the number of “-1” values in the Kansei vectors of the peers is still larger than the number of “+1” values and the p value of the peers is equal to the number of “-1” values. However, when using prototypical Kansei vector (1) and a small value of p_b , since the numbers of “-1” and “+1” values in the Kansei vectors of the peers are almost the same, the Kansei query forwarding method does not produce positive feedback to specific peers. On the other hand, even when using prototypical Kansei vector (1) or (2), the

one-walker random walk is expected to provide high search performance that is equivalent for all peers, because each peer has the same tendency with respect to file collection. Therefore, each peer would basically have files that other peers desire. Finally, even when prototypical Kansei vector (1) or (2) is used, if the value of p_b becomes larger, the numbers of “-1” and “+1” values in the Kansei vectors of the peers become more similar and randomness is needed in the file search. Therefore, the abilities of the Kansei query forwarding method and the one-walker random walk to induce convergence would become more similar with increasing p_b . The simulation results are summarized in the following.

- Case of random determination of p

If the Hamming distance between the Kansei vectors of participants is small (the similarity between the Kansei information of the participants is high) and the difference between the numbers of “-1” and “+1” values in the Kansei vectors of participants is small (the bias of impression to distributed files is small), the Kansei query forwarding method has a better ability to induce convergence. Otherwise, there is almost no difference between the Kansei query forwarding method and the

one-walker random walk in terms of the ability to induce convergence.

- Case of Kansei-correlated determination of p

If the Hamming distance between the Kansei vectors of participants is small (the similarity between the Kansei information of participants is high) and the difference between the numbers of "-1" and "+1" values in the Kansei vectors of participants is large (the bias of impression to distributed files is large), the one-walker random walk has a better ability to induce convergence. Otherwise, there is almost no difference between the Kansei query forwarding method and the one-walker random walk in terms of the ability to induce convergence.

6. CONCLUSIONS

In the present paper, we proposed the concept of P2P file sharing networks that allow participants (peers) to freely assign structured meta-data to their own files, that is, we proposed P2P file sharing networks that allow participants to group files in their own ways. We focused on vectorized Kansei information as a concrete example of such structured meta-data. The vectorized Kansei information indicates what participants feel to their own files, and search queries take the same form as the vectorized Kansei information. In addition, as a mechanism that enables file search using the vectorized Kansei information as meta-data, we proposed the Kansei query forwarding method. The Kansei query forwarding method is meant to propagate a search query among peers that have similar Kansei information. The proposed Kansei query forwarding method measures the similarity between a search query and meta-data that is assigned to the files of peers by calculating their dot products.

In the simulation experiments, we examined whether the Kansei query forwarding method and, for the purpose of comparison, the random-walk-based query forwarding method can provide equal and high search performance for all of the peers in the network in the situation in which the Kansei information of the peers and the tendency of the peers with respect to file collection are diverse. Although the models of the Kansei information and the tendency for file collection mentioned herein are simple and it is uncertain as to whether the models are a sufficient reflection of the real-world environment, the simulation results indicated that the Kansei query forwarding method and the one-walker random walk provide equal search performance for all of the peers in different situations under the use of the models. Furthermore, it was shown that these two query forwarding methods are complementary. Meanwhile, in the present paper, we do not consider the difference in popularity among files, the variety of network topologies, etc., that would cause inequalities among participants in terms of search performance, and these factors should be taken into consideration in future studies.

The motivation behind the present work was to enable communication between participants in which information is intermediate. In the proposed P2P file sharing networks, participants look for other participants who have a similar sense of file grouping, as well as the files themselves, through a file search process. The simulation results presented in this paper suggest the possibility that participants with similar

Kansei information can exchange files with each other, and the abovementioned communication would be one step of the file exchange process described herein. In future studies, we will develop a P2P file sharing network along with the concept introduced in the present paper and will verify whether such communication is possible through file search in the P2P network.

Acknowledgment

This work was supported in part by the Ministry of Internal Affairs and Communications, Japan, and in part by the Japan Society for the Promotion of Science, Grant-in-Aid for Scientific Research (S)(18100001).

7. REFERENCES

- [1] E. Cohen and S. Shenker. Replication strategies in unstructured peer-to-peer networks. In *Proceedings of the ACM SIGCOMM 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, Pittsburgh, PA, USA, August 2002.
- [2] S. Joseph. Neurogrid: Semantically routing queries in peer-to-peer networks. In *Proceedings of the International Workshop on Peer-to-Peer Computing (co-located with Networking 2002)*, Pisa, Italy, May 2002.
- [3] S. Joseph and T. Hoshiai. Decentralized meta-data strategies: Effective peer-to-peer search. *IEICE Transactions on Communications*, E86-B(6):1740–1753, 2003.
- [4] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys & Tutorials*, 7(2):72–93, Second Quarter 2005.
- [5] Q. Lv, P. Cao, E. Cohen, and S. Li, K. Shenker. Search and replication in unstructured peer-to-peer networks. In *Proceedings of the 16th international conference on Supercomputing*, pages 84–95, New York, USA, June 2002.
- [6] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: Scalable peer-to-peer lookup service for internet applications. In *Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 149–160, San Diego, CA, USA, August 2001.
- [7] H. Takagi and T. Noda. Media converter with impression preservation using a neuro-genetic approach. *Hybrid Intelligent Systems*, 1(1):49–56, 2004.
- [8] C. Tang, Z. Xu, and S. Dwarkadas. Peer-to-peer information retrieval using self-organizing semantic overlay networks. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM'03)*, pages 175–186, Karlsruhe, Germany, 2003.
- [9] K. Yoshida, T. Kato, and T. Yanaru. A study of database system with kansei information. In *Proceedings of 1999 IEEE international Conference on System, Man, and Cybernetics*, pages 253–256, 1999.