A Bloom Filter-Based User Search Method Based on Movement Records for P2P Network

ARTICLE · JANUARY 2009

DOI: 10.1109/SAINT.2009.41

5 AUTHORS, INCLUDING:

Yuuichi Teranishi
Osaka University
53 PUBLICATIONS **114** CITATIONS
SEE PROFILE

Kaname Harumoto
Osaka University
56 PUBLICATIONS **167** CITATIONS
SEE PROFILE

A Bloom Filter-Based User Search Method Based on Movement Records for P2P Network

Toru Shiraki †, Yuichi Teranishi †‡, Susumu Takeuchi ‡, Kaname Harumoto ††, Shojiro Nishio †
†Graduate School of Information Science and Technology, Osaka University
‡National Institute of Information and Communications Technology
††Graduate School of Engineering, Osaka University
shiraki.toru, teranisi, nishio@ist.osaka-u.ac.jp
stakeuti@nict.go.jp, harumoto@eng.osaka-u.ac.jp

Abstract

In this paper, we propose a P2P user search method based on movement records that can be automatically obtained by location detection devices. We assume movement records are treated as a sequence of pairs of spot-ID and time and they are stored in a peer for each user.

In our proposal, a Bloom Filter is applied to each spotID and time to combine all movement records for one user as a fixed length bit array. To search a user who followed specified course, we propose a AND/OR search method based on Bloom Finger Table (BFT), which extends a routing table of a Chord DHT system to retrieve elements using Bloom Filter. By this method, user searches based on a sequence of locations with or without time can be realized efficiently.

Additionally, in order to reduce the number of messages for a user search, we propose a peer-ID assignment for BFT based on user's geographical foothold. The number of messages for a user search can be reduced by this peer-ID assignment since users who visit same places are located closer to each other on the routing table. Evaluation results of simulations show that our proposal reduces the number of messages compared to a naive implementation using existing P2P retrieval method.

1. Introduction

Recent technological advances on location detection devices such as GPS and ubiquitous networks made it possible for mobile users to obtain location-based information services that utilizes user's current geographical position.

In addition, researches on location-based information services that utilize movement records in the past are now in progress. Examples of such services are, a sightseeing navigation service that recommends next destination by searching movement record of another user who visited same places [4], a messaging service to the people who attended to the same conference, and so on.

To realize such services, user search mechanism that can retrieve users who followed specified course is required. We assume user's movement records are stored in the independent storages that are managed by independent organizations or individuals. Therefore, we assume to manage movement records in the distributed peer-to-peer (P2P) environment.

The goal of our research is to realize effective user searches based on user's movement records that are shared in the P2P network. We assume movement records are treated as a sequence of pairs of visited location (represented as spot-ID) and time. To treat such user searches based on movement records, we propose a movement record management method that combines multiple movement records to a fixed length bit array. To be concrete, we apply *Bloom Filter* [1] for management of movement records. Bloom Filter is a simple space-efficient randomized data structure for representing a set in order to support membership queries. There are some researches on search method based on Bloom Filter for P2P networks. Bloom Finger Table (BFT) [3], which extends a routing table called Finger Table used in Chord [5] is one of them.

We propose a user search method based on realize movement records, which extends BFT. In our proposal, two kinds of Bloom Filters, one is generated from visited spot-IDs and other is generated from corresponding time, are generated and concatenated. By this Bloom Filter structure, user searches based on a sequence of locations with or without time can be realized efficiently. Additionally, in order to reduce message number for a user search, we propose a peer-ID assignment for BFT based on user's geographical foothold. The number of messages for a user search can be reduced by this peer-ID assignment since users who visit

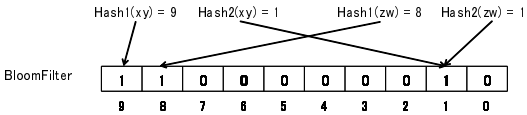

Figure 1. Bloom Filter

same places are located closer to each other in the Chord network.

2. Bloom Finger Table (BFT)

We extend Bloom Finger Table (BFT), which extends Chord for Bloom Filter searches, to realize user search based on movement records. Firstly Bloom Filter and BFT are briefly explained in this section.

Bloom Filer is a simple space-efficient randomized data structure for representing a set in order to support membership queries. To create Bloom Filter, firstly bit array with length *m* have to be initialized as '0'. To add an element, feed the element to each of the *k* hash functions that returns *0* to *m-1* to get *k* array positions and set the bits at all these positions to '1'. For example, in the Figure 1, *k=2* and two bits are set. In this case, element *xy* and *zw* are added and corresponding bits are set as '1'. To test whether an element is in the set, feed the element to each of the *k* hash functions to get *k* array positions. If any of the bits at these positions are '0', the element is not in the set. If all are '1', then the element may be in the set. The test of the Bloom Filter has false positive.

BFT applies the Bloom Filter to the Finger Table of the Chord. For example, in the Figure 2, the finger table of Peer 0 contains Bloom Filter of peer 0, peer 1, combined Bloom Filter generated from Bloom Filters of peer 2 and peer 3, from peer 4 to peer 7, and from peer 8 to peer 15. To search multiple elements, a combined Bloom Filter of the elements is generated. Then the query is forwarded to the peer in the Finger Table when the Bloom Filter of the query is included in the corresponding Bloom Filter. In the example of Figure 2, the query is forwarded to the peer 4 since the Bloom Filter is included. If the Bloom Filter of the peer is included by the Bloom Filter of the query, the peer is matched to the query. However, all elements have to be tested again on the peer since the testing by Bloom Filter has false positive. If peer 4 does not include all elements, then the query is ended. Otherwise, the same query process is repeated from the peer 4.

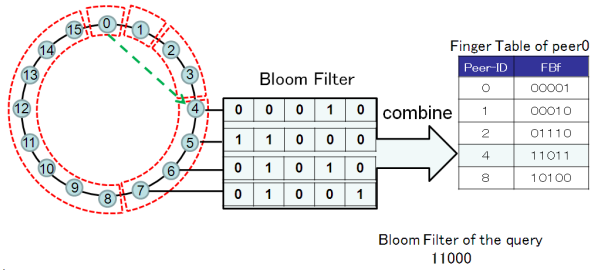

Figure 2. Bloom Finger Table

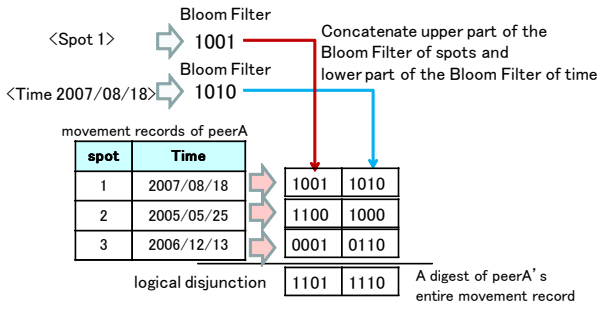

Figure 3. Movement Record Management

3. User Search Method Based on Movement Records

3.1 Movement Record Management on BFT

In this paper, we propose a user search method based on movement records by extending BFT.

In our proposal, as shown in the Figure 3, we apply Bloom Filter to each spot-ID and corresponding time independently. Then concatenate them as a one Bloom Filter. This Bloom Filter indicates one movement record. In our assumption, one peer corresponds to one user. Therefore, logical disjunction (OR) of all Bloom Filters on the peer represents a digest of one user's entire movement record. By simply applying this Bloom Filter structure to the BFT, user search based on the movement records can be realized.

In this Bloom Filter structure, upper part of the Bloom Filter represents a digest of spots which user visited. Therefore, if a query issuer need not limit the visited time, then each BFT peer only have to ignore the lower part of the Bloom Filter.

3.2 AND Search

'AND search' corresponds a query to retrieve all peer-IDs of the peers that include all of the specified multiple

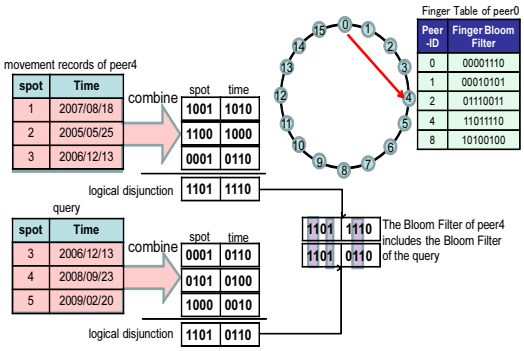

Figure 4. AND Search

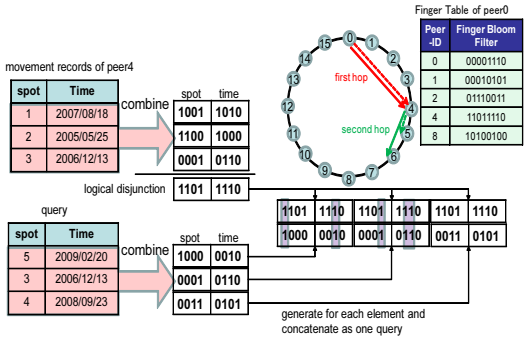

Figure 5. OR Search

pairs of spot and time (multiple movement records) in the peer's movement records. For example, if a query issuer wants to retrieve users who followed specified course at specified time, then the query becomes AND search.

Basically the AND search is same as the original BFT. To search with multiple movement records, applying logical disjunction of the Bloom Filters of the movement records is generated as a query. Then the query is forwarded with the process described in the former section. If a query issuer don't want to limit the visited time, then each BFT peer only ignore the lower part of the Bloom Filter.

In the Figure 4, the movement records of the peer 4 are converted to one Bloom Filter by logical disjunction. The generated Bloom Filter is stored in the Finger Table of the peer 0. In this example, the query consists of three movement records. In this example, the query is forwarded from peer 0 to peer 4 since the Bloom Filter of peer 4 includes the Bloom Filter of the query. This query forwarding process is repeated until the Bloom Filters in the Finger Table does not include the Bloom Filter of the query. At the forwarding time, all Bloom Filters of the movement records are compared to the Bloom Filter of the query. If the Bloom Filter of the movement record includes the Bloom Filter of the query, then the peer-ID and the corresponding movement record are returned to the peer of the query issuer. But since Bloom Filter has false positive, the query issuer have to check the movement record again.

3.3 OR Search

'OR search' corresponds to a query to retrieve all peer-IDs of the peers that includes one of the specified multiple pairs of spot and time (multiple movement records) in the peer's movement records. For example, if a query issuer wants to retrieve users who followed any of the specified courses at specified time, then the query becomes OR search.

In the OR search, Bloom Filters are generated for each element and concatenated as one query. If one of the each element of the Bloom Filter is included, the query is forwarded to the peer. In the Figure 5, if one of movement records of the peer5 is spot=3 and that of the peer6 is spot=5, peer4 includes the query. Therefore, the query is forwarded to peer4 at the first hop. But at the second hop, peer5 and peer6 include the query and two queries are forwarded to them.

3.4 Geographical Assignment of Peer-IDs

In Chord, peer-IDs are assigned randomly to the peers. Therefore, if many users visited to the same place at same time, the query for the visit spreads to many peers even in the above method. In our proposal, in addition to the BFT, we propose a peer-ID assignment based on user's geographical foothold. In the real life, if the footholds of the users are close to each other, then the movement records becomes similar since user's movement area is generally not so wide. Therefore by assigning the peer-ID according to the footholds of the user (e.g. the position of the user's home), the users with closer peer-ID tend to have similar visited places, which means the Bloom Filter of the BFT also becomes similar. If the Bloom Filters are similar, then the logical disjunction of the Bloom Filters also becomes similar. Hence, the queries are not spread to many peers in the BFT. In the Figure 6, since peers from 4 to 7 have similar movement records, the queries for these movement records only have to be forwarded to the peer 4 from peer 0. This reduces the number of message for forwarding.

4. Evaluation

4.1 Simulation Setup

To evaluate the effectiveness of our proposal, we implemented a simulator and evaluated the proposed method with

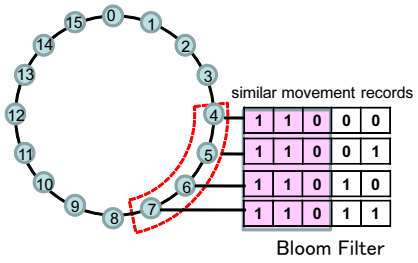

Figure 6. Geographical Peer-IDs

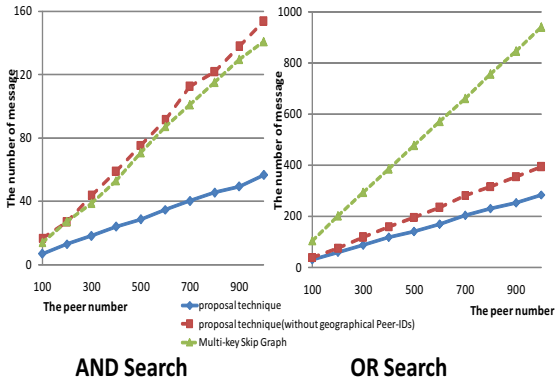

Figure 7. The Number of Message

simulation. We compared our method with a naive implementation using existing P2P retrieval method called Multi-key Skip Graph (MKSG) [2], which can handle multiple element retrieval.

In our simulation, the length of the Bloom Filter is set as 4096. The number of hash functions (k) is set as 1. Each peer has 100 movement records. There are 1000 spots and each user have 5 same movement records when they have neighbor peer-ID. 'time' is represented as a number of days since UNIX epoch time. The max number of the 'time' is set as 14925 (2009/2/20) and randomly assigned to each movement record. We set peer-ID from 0 to 1000 to each peer and the peer-ID neighboring peers have 60% same movement records. This corresponds to setup that the foothold neighboring users have the 60% same visited places by assigning geographical peer-ID. The query includes 5 movement records and 5 % of the peers have the movement records in the AND search and 10 % of the peers have the movement records in the OR search.

4.2 Simulation Result

Figure 7 shows the result of the correlation between the number of messages in AND search and the number of the peers. The number of message is represented the average of the number of messages when a query is forwarded. The number of messages becomes almost same as MKSG when the proposed method does not use geographical peer-ID. But with the geographical peer-ID, the number of message is vastly improved (Reduced 56 %). That is the spread of the query was suppressed. Figure 7 shows the result of the number of messages in OR search changing the peer numbers. As shown in this figure, proposed method with or without peer-ID reduces the number of messages compared to MKSG (reduced 64 %). In MKSG, when we retrieve multiple movement records, multiple queries are necessary. However, because they are concatenated with one by our propose, the number of message was suppressed.

5. Conclusion

In this paper, we proposed a Bloom Filter-based user search method based on movement records in the P2P architecture. By applying Bloom Finger Table and geographical peer-ID assignment, the number of messages is vastly reduced compared to the naive implementation.

In the future work, the overhead of constructing Bloom Finger Table should be reduced. In addition, another experiment performed by real users should be evaluated for verifying the effectiveness of geographical peer-ID assignment.

Acknowledgements

This research is partly supported by the Ministry of Internal Affairs and Communications of Japan under the Research and Development Program of 'Ubiquitous Service Platform (2008)'.

References

[1] B. Bloom, Space/time trade-offs in hash coding with allowable errors: Communications of the Association for Computing Machinery, 13(7), pp.422-426, July 1970.

[2] Y. Konishi, M. Yoshida, S. Takeuchi, Y. Teranishi, K. Harumoto, S. Shimojo, An Extension of Skip Graph to Store Multiple Keys on Single Node(In Japanese): Information Proceeding Society of Japan, vol.49, No.9, pp.3223-3233, Sept. 2008.

[3] K. Sato, N. Matsumoto, N. Yoshida, Multi-Keyword Search for DHT P2P Networks: FIT, CD-ROM, Sept. 2007.

[4] H. Shinoda, S. Takeuchi, Y. Teranashi, K. Harumoto, and S. Shimojo, A ubiquitous activity navigation method by collaborative filtering based on behavioral characteristics: Proc. of International Conf. on Computers in Education, pp.297-302, Nov. 2007.

[5] I. Stoica, R. Morris, D. Kargery, M. F. Kaashoek, and H. Balakrishnan, Chord: A scalable peer-to-peer lookup service for internet applications. SIGCOMM Comput. Commun. pp.149-160, Aug. 2001.