Proposal of Decentralized Information Sharing System using Local Matchmaking

Masahiro Hamasaki
Graduate Universities of Advanced Studies
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo JAPAN
Email: hamasaki@grad.nii.ac.jp

Hideaki Takeda
Graduate Universities of Advanced Studies
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo JAPAN
Email: takeda@nii.ac.jp

***Abstract*— In this paper, we propose a re-configuration method for personal human network and show the information sharing system using it. Personal human network is useful for various utilization of information like information gathering. Since a personal network is usually ad-hoc network, it is necessary to extend and optimize it. Using the neighborhood matchmaker method, we can increase a new friend who is expected to share interests via all own neighborhoods on the personal human network. Iteration of matchmaking is used to optimize personal human networks. We simulate the neighborhood matchmaker method and compare the results by our model with those by the central server model. The neighborhood matchmaker method can reach almost the same results obtained by the sever model.**

I. Introduction

We now live so-called ”Information flood era”, i.e., we can obtain enormous information that we have never in other era. The trouble is that there is too much information to select and use adequately. Search engine can help us but they are limited in quality of search engine types of information. Is any other help available? The great help is of course people.

Information exchanging among people is one of powerful and practical ways to solve information flood because people can act intelligent agents for each other to collect, filter and associate necessary information. The power stems from personal human network. If we need variable information, we must have a good human network.

Personal human network is useful for various utilization of information like information gathering, but it is usually formed locally and often indepedently. In order to adapt various needs for information utilization, it is necessary to extend and optimize it. In this paper, we propose a network optimization method called ”*Neighborhood Matchmaker Method*” and show the prototype system using it. It can optimize networks distributedly from the arbitrarily given networks.

II. Related Work

There are some systems to capture and utilize personal human network in computers. Kautz et al. [1] emphasized importance of human relations for WWW and showed done primary work for finding human relations, i.e., their system called ReferralWeb can find people by analyzing bibliography database. Sumi et al. [4] supported people to meet persons who have same interests and share information using mobile computers and web applications. Kamei et al. supported to form communities using visualization relationship among participants [2]. SoMeOne is the system to find people that act as relevant information source [3]. It regards persons as contents.

In these systems, they assume target groups either explicitly or implicitly. This arises the following two problems as internet community support systems. The first problem is how to form such groups, especially how we can find people as members of groups. We call it ”meet problem”. The second problem is how to find suitable people in groups for the specific topics and persons. We call this problem ”select problem”. The bigger group contains more likely valuable persons to exchange information. However, we have to make more efforts with these systems in order to select such persons from a lot of candidates in the group. It is difficult for us to organize and manage such the large group.

Therefore information exchanging systems should support methods that realize the above two requirements i.e., to meet and select partners.

III. Neighborhood Matchmaker Method

A. Proposal of Local Matchmaking Method

As we mention in the previous chapter, if we need better relationship for information exchanging, we must meet and select partners more and more. It is a big burden for us, because we should meet all the candidates before we select them in advance. Since we do not know new friends before meeting them, we have no ways to select them. How can we solve this problem in our daily life? The practical way is introduction of new friends by the current friends. It is realistic and efficient because the person who knows both can judge whether this combination is suitable or not. Friends work as matchmaker for new friends. We formalize this ”friends as matchmaker” as an algorithm to extend and optimize networks.

The key feature of this approach is no need for central servers. The benefits of this approach are threefold. The first is to keep spread of information minimally. Information on a person is transferred to only persons connected to her/him directly. It is desirable to keep personal information secure. The second is distributed computation. Computation to figure out better relationship is done by each node, i.e., computers used by participants work for it. It is appropriate for a personal

0-7803-7978-0/03/$17.00

human network because we do not have to care the size of network. The third is gradual computation. The network will be converged gradually so that we can obtain the optimal network to some extent even if we stop the computation anytime.

B. Formalization

We introduce a model that can optimize networks by formalizing the method in our real life. We call this method *"Neighborhood Matchmaker Method (NMM)"* hereafter. Before explaining *NMM*, we define the network model for this problem. At first we define a person as a node, and a connection for information exchanging between people as a path. Here we assume that we can measure a degree of the connection between two nodes (hereinafter referred to as "connection value"). Then, we can define that making a good environment for information exchanging is optimizing this network. In *NMM*, the network is optimized by matchmaking of neighbor nodes.

We need the following two conditions to apply *NMM*.

- All nodes can possibly connect to each other
- All nodes can calculate relationship between nodes connected to them

Under these conditions, all nodes can act as matchmakers for their connected nodes to improve the connection network. The behavior of a node as a matchmaker is as follows.

1) A node calculates connection values between its neighbor nodes. (We call this node "matchmaker")
2) If the matchmaker finds a pair of nodes that has a good enough connection value by computation of connection values, it selects this pair for recommendation. Then the matchmaker introduces both nodes of recommended pair to each other.
3) The node that receives recommendation decides whether it accepts or not.

We can optimize personal human network by iteration of this behavior. Figure 1 shows these behaviors.

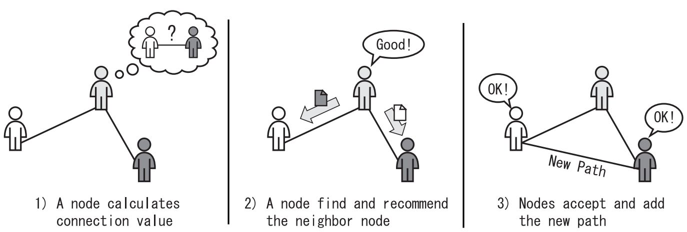

Fig. 1. Behavior of nodes

IV. System Overview

In this chapter, we introduce the prototype system using *NMM*. This system supports us to share URLs and comments. When you discover some web pages that you are interested in, you often keep it with some comments in memorandum, or register it to web bookmarks in your web browser. Moreover, you may recommend it to your friends. You may be recommended some web pages by your friends to the contrary. This is one of examples to exploit personal human networks for information utilization. This system supports these activities.

Figure 2 shows the overview of this system. All users have their own personal server that is a CGI program. When we submit URLs and comments to own server through your web browser, the server stores them in the database. And then, it sends them to neighborhoods. In the same way, the neighbors send such information to the system, and it stores them. A neighbor is a person who contracts to exchange information with us. We need to set the first neighbors in order to exchange information. [1]

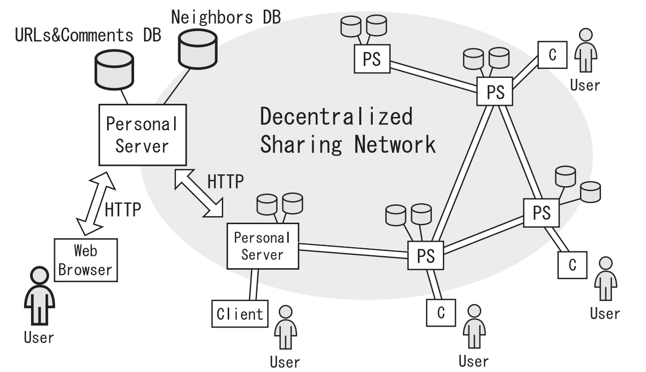

Fig. 2. System overview

Using this system, we can see the information both uploaded by themselves and sent by neighbors. Moreover, the system calculates the connection values among neighbors based on the information sent by them. It forwards URLs and comments to pairs of neighbors that have high connection values and introduce each other automatically. We can extend the personal human network through the each system does such intermediation activity.

We adopt the following policy for information sharing for the security reason.

- URLs can be shared by everyone.
- Comments to URLs can be shared by people who are reliable for the comment writer.

Reliability is represented as neighborhood in the personal network in this system. We explain this situation with Figure 3. User A finds the URL and comments to it (1). User B receives the URL and comments to it (2). Then, the both of User A and B share the URL and comments written by each other but User C only share the URL because of the policy in this system (3). When User C comments to the URL, User B can receive the URL and comments from User C (4). As a result, User B can have both of comments from User A and C to the same URL (5).

In order to use *NMM*, the system should be able to calculate connection values among neighbors. We regard the pair of neighbors has the same interests if both of the pair write comments about the same URLs. In this system, the connection value is the number of the pair of the comments about same URL. If the number of such comments is higher than the

[1] In order to contact the first neighbors, we have to use e-mail or something else to communicate. Because there is no participants list since it is the decentralized system.

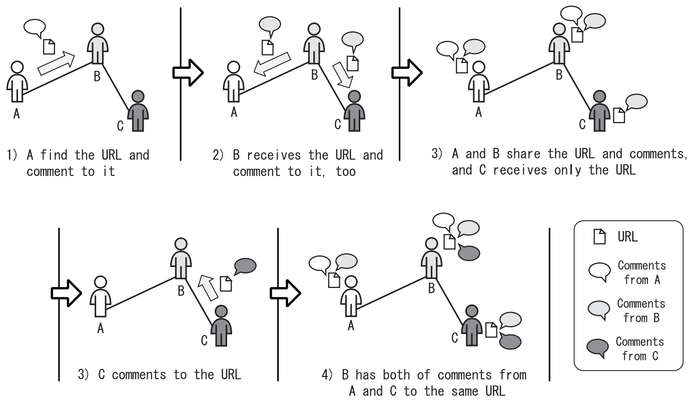

Fig. 3. Sharing model

threshold, the system regards they has a good relationship and introduce a pair of neighbors to each other. And then we can form personal networks.

The probability of the same URL found and wrote comments by the neighbors independently is little because there is enormous URLs on the Web. In this system, URLs that a user registers can be transferred to other users through the personal network. Then there are chances for multiple users to comment the same URLs.

V. Experiments

Since *NMM* just ensures local optimization, we should investigate the global behavior when applying this method. We test the method by simulation. We simulate optimization with *NMM* using the random data.

A. The Procedure of the Simulation

In the chapter III-B, we introduce *NMM* as the three steps, but the third step, i.e., decision for accepting recommendation is free to choose. In the simulation, we use a simple tactics. Each node wants to connect to other nodes that have better connection values i.e., if a new node is better in connection than the worst existing node, the former replaces the latter.

We explain how simulation is performed.

At first we should prepare initial states for nodes. We create nodes each of which has data to represent a person. In this experiment, a node data is a ten dimensional vector. We initially put paths between nodes randomly. We fix the number of paths during simulation. It means that addition of a path requires deletion of a path. Simulation is iteration of calculation for recommendation and exchanging of paths based on the calculation.

All nodes calculate connection values among own neighbors. The connection value is a product of two vectors in this experiment.

In this simulation only one node can exchange paths in every turn. The node must add the best path and remove the worst path instead. So that, the number of paths in the network is fixed. The adding path must be better than the worst path already had. If all nodes cannot get a new path using matchmakers, the network is converged. At that time, this simulation is terminated.

B. The Measurement

Since the purpose of the simulation is how our method achieves optimization of the network, we should define what is the optimized network. We adopt a simple criterion. The best network for *n* paths is the network that includes *n* best paths in connection values [2]. Good news is that this network can be easily calculated by collecting and computing information for all nodes. Then we can compare this best network and networks generated by our method. Of course this computation requires a central server while our method can be performed distributedly. In other words, we compare central server model versus distributed model (*NMM*).

We compare two networks in the following two ways. One is cover rate. It is how many paths in the best network are found in the generated network. It means how similar in structure two networks are. The other is reach rate that is comparison of the average of connection values between the best and generated networks. It indicates how similar in effectiveness two networks are. These parameters are defined as the following formulas:

$$cover\ rate = \frac{|\,\{P_{current} \cap P_{best}\}\,|}{N}$$

$$reach\ rate = \frac{\sum_{l=1}^{N} f(p_l | p_l \in \{P_{current}\})}{\sum_{m=1}^{N} f(p_m | p_m \in \{P_{best}\})}$$

p	:	a path
N	:	the number of paths
$\{P\}$	:	a set of paths
$\{P_{best}\}$	:	the best set of paths
$\{P_{current}\}$	:	the current set of paths
$f(p)$	:	a value of path

C. Results

In this simulation, we should provide data that can be used to calculate connection values between nodes. As we mentioned we define that each node has a ten dimensional vector generated randomly, and the connection value between nodes is product of two vectors. An element of a vector is a random number from zero to one. The probability for which zero is set on the element is 0.5. The reason is to make heterogeneous relationship among nodes.

There are two parameters to control experiments. One is the number of nodes and the other is the number of paths. In this experiment, we vary the number of nodes from ten to one hundred and the number of paths from the one to five times the number of nodes. The simulation is performed ten

[2] This criterion may not be ”best” for individual nodes, because some nodes may not have any connections. We can adopt other criterion if needed.

times for each set of parameters, and we use the average as the results.

The graphs in Figure 4 and Figure 5 plot the average of cover-rate against turn. Figure 4 shows the results varying the number of paths and fixing the number of nodes, and Figure 5 shows the results by varying the number of paths and fixing the number of nodes.

In our formalization, we cannot know whether the network will converge. However, we can see that all graphs became horizontal. It implies that all networks were converged using matchmaking.

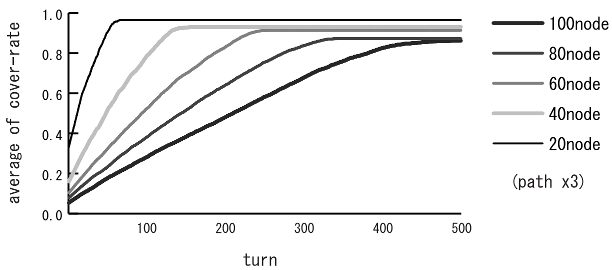

Fig. 4. Cover-Rate and nodes

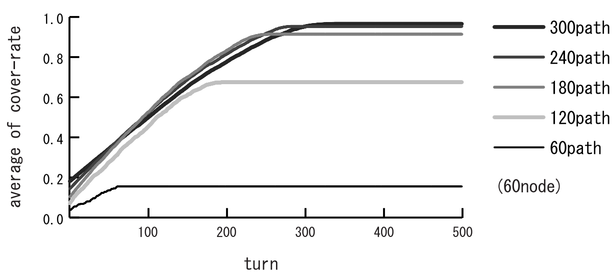

Fig. 5. Cover-Rate and paths

We observed similar results on reach-rate. The difference of reach-rate is less dependent on the number of paths and nodes.

We also examine the relevance between the number of networks and the turn of convergence. The graph in Figure 6 plots the average of convergence turns against the number of nodes. This graph indicated that the turn of convergence increases linearly when the number of nodes increases. In this simulation, only a single node can exchange paths in a turn, so the times of exchanging per node is irrelevant to the number of nodes.

Let me estimate the complexity of computation of the algorithm roughly. When the average of the number of neighborhood nodes is r, this algorithm calculates connection values $2r$ times in every turn. When the number of nodes is N and the number of turns of convergence is kN according to Figure 6, the calculation times to converge is $2rkN$ using *NMM*. In the centralized model the calculation times is N^2 because we have to calculate connection values among all nodes. Since r and k are fix value, the order is $O(N)$ using *NMM*. It is less than $O(N^2)$ using the centralized model.

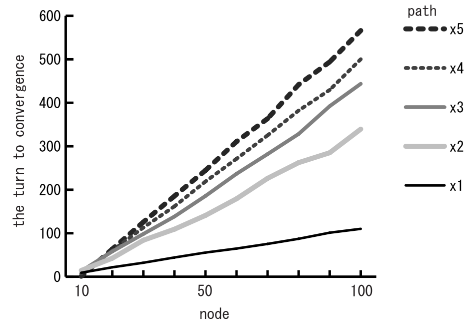

Fig. 6. Average of Convergence Turn

VI. Conclusion

In this paper, we propose a method called *"Neighborhood Matchmaker Method (NMM)"* as the way to get a new person who is a partner for exchanging information and show the information sharing system using it. Our method only use collaborative and autonomous matchmaking and do not need any central servers.

It is applicable to any size of community, because it calculates relationship among people without collecting all data. It is possible to assist bigger groups that are more likely to contain valuable persons to exchange information. And it is less computational cost. Furthermore it is an easy and quick method because we can start up anytime and anywhere without registration to servers. We can assist to form dynamic and emergent communities that are typical in the Internet.

We show the prototype system using the proposed method. In the real world, personal network changes dynamically through the exchanging information among people. A further direction of this study will be to experiment with this system and investigate effectiveness for it in real world.

References

[1] Henry Kautz, et al., ReferraWeb: Combining Social Networks and Collaborative Filtering. In *the Communications of the ACM*, Vol. 40, No. 3, 1997.

[2] Koji. Kamei, et.al., Community Organizer: Suppoting the Formation of Network Communities throuh Spatial Representation. In Proceedings of *the 2001 Symposium on Applications and the Internet (SAINT01)*, 2001.

[3] Michel PLU, et al., The Web of People: A dual view on the WWW. In *Proceedings of The 12th International World Wide Web Conference (WWW-12)* (2003).

[4] Yasuyuki Sumi, et al., Collecting, Visualizing, and Exchanging Personal Interests and Experiences in Communities. *the 2001 International Conference on Web Intelligence (WI-01)*, 2001.