High Assurance Communication Technique for Autonomous File Sharing Community

Khaled Ragab, Takanori Ono and Kinji Mori
Tokyo Institute of Technology
2-12-1 Ookayama, Meguro, Tokyo 152-8552
Tel: +81-3-5734-2664, Fax: +81-3-5734-2510,
Email: {ragab@mori., nouri@mori., mori@}cs.titech.ac.jp

Abstract

The headway towards the outstanding future can be realized with the productive cooperation among peoples and organizations. The productive cooperation has high demands for providing reliable and efficient information and function sharing. For achieving these requirements we propose an **Autonomous community** (**AC**) concept. **AC** is a group of autonomous members, in which each member has his own objectives, complies with the community obligations and has to be mutually cooperative with the others for achieving his own objectives. For achieving high reliable communication among community members we proposed a community communication technique based on one-to-many (1->N). This technique is characterized by **by-productivity** property. This property assures that not only the requester will receive the shared information but also the other members receive this information without sending the same request.

Keywords: Assurance, Autonomous Community, Autonomous Decentralized Systems and Reliable Communication technology.

1. Introduction

Day by day both the number of the Internet users and web pages are growing extremely. There were 459 million people with Internet access in the 30 countries studied by Nielsen NetRatings at the end of 2001. The predicted number of people that will access the Internet by 2005 is 1.17 billion [9]. Moreover the Internet users demands for achieving their services in short time increased within this century as a result of our dynamic life characteristics. The Internet users submit their search requests to a search engine. Thank to remarkable search engines such as **Google, AltaVista, Infoseek**, etc. They search millions of web pages around the world and reply in a few seconds by a list of pages that match the user request. This list of pages consumes a lot of users time for getting their information. So, there is a high demand for Internet facilities that provide the Internet services to the Internet users efficiently with low efforts. File sharing is one of the Internet services. Napster [4], and Scour Exchange [5] are two centralized Internet applications used mainly for file sharing. Napster is a system to share mp3 files with centralized database of resources and users. Secure Exchange is a system similar to Napster, except it is not limited to operate with mp3 files. While Gnutella [6] is fully decentralized one in which users share their interest of the data files..

Inspired from the spirit of cooperation in the social communities, we propose a promising Internet system design locally based, locally driven communication that simulate the cooperation activities on a community. The major aim of the community is to form a group of users and service providers for efficient function sharing and information sharing. Our goal is to provide an efficient interaction among community members who share common interests, relationships, and fantasies, as well as to members who seek to buy and sell products, services or information. Moreover community is very important for making decisions in our daily life: where to go, what to buy, what to do all near to us. Whether we are looking for businesses, services, products or activities community can offers comprehensive, high-quality local editions with thousands of local event listings and restaurant reviews, city guides for residents and visitors and much more.

The major focus of this paper is autonomous community reliable communication techniques. This technique is the core of our proposed Autonomous community concept, [3]. The remainder of the paper is organized as follows. The next section presents the requirements of both the Internet users and the Internet file sharing systems. While in section 3 briefly we present the autonomous community definition and concept. In section 4, we introduce the communication technology for information sharing. Section 5, proves that our communication technology is reliable. The last section draws conclusions.

2. Requirements

The Internet users have a wide and different objectives and interests that change frequently under the evolving situations. The Internet file sharing systems should tackle the Internet users requirements by introducing new facilities, and new concepts.

2.1 Users Requirements

The Internet users requirements can be briefly listed as follows:

- They want to have a one-click response for achieving their services. Moreover, they are keen to achieve their services with low efforts and high benefits.
- The Internet users aspire to active cooperation among them.

- They are often changing their interests, objectives and businesses.
- They always have demands for highly quality services.

2.2 System Requirements

The Internet file sharing systems requirements are as follows:

- **Timeliness**. The systems have to provide the services easily, in short time and comfortable to the Internet users.
- **By productivity**. They must provide efficient ways for the productive cooperation among the Internet users
- **Flexibility.** The systems have to provide some efficient techniques to deal with the requirements of the Internet users that change frequently.
- **Reliability**. They must assure that the communication among the Internet users is reliable. In this paper we will introduce the reliability model.

These requirements shove us for proposing a new system based upon the community spirit. In order to satisfy these requirements mentioned before and inspired from the Autonomous Decentralized System (ADS) concept [1][2], we propose an Autonomous community concept. In the next section briefly we will present our proposed Autonomous community definition and concept.

3. Autonomous Community

Autonomous Community is a group of autonomous members, in which:

- **Each member has his own objectives**
- **Each member complies with the community obligations.**
- **No member controls the flow of information to the other members.**
- **The community members have to be mutually cooperative for achieving their own objectives.**
- **The community membership burdens must be less than its benefits.**

Community member objectives and community obligations change based on the application. The community members can achieve their own objectives efficiently and more easily in comparing without joining the community.

3.1. Autonomous Community Concept

The community must satisfy the following aspects to be autonomous community:

1. **Autonomous Coordinability.** If any member leaved, his node failed, or new member joined the community, the other community members can coordinate their individual aspirations among themselves and each member can operate in a harmonious (coordinated) fashion.
2. **Autonomous Controllability.** If any member leaved, his node failed, or new member joined the community, the other community members can continue to manage themselves to commit their own tasks.
3. **Community Commitments.** Each member in the community has to comply with the community commitments. Each member has to pay the community fee. The community fee changes based on the community application. For example in file sharing community, each member is required to provide some files for sharing in order to make the community alive.
4. **Community's Mutual Benefits.** The constructive cooperation among members is the most important feature of the autonomous community. Moreover, the community members can reap the useful information for achieving their objectives with low efforts and high benefits. The

community benefits have to cover its liability to be attractive for both its members and the other users to join it.

These aspects assert that the community structure changes dynamically, each member has autonomy for interactive communication and information processing. Moreover, the community members cooperate for utilization and provision of the community services and information sharing under the evolving situations.

The autonomous community concept can be realized with autonomous controllability and autonomous coordinability. Moreover, each member is required to satisfy the following conditions:

- **Equality**. Each member must be equal and can handle his objectives without being directed by or giving directions to the others. There is no community member control the information flow of the community to the other members.
- **Locality.** Each member must handle his objectives and coordinate with the others based only on the local information.
- **Synergy.** The productive cooperation among community members is achieved by using efficient communication techniques. Community members perform communication on a one-to-many (1->N) basis.

4. Autonomous Community Communication Technology

4.1 Autonomous Community Network

The autonomous community network logical topology is a set of nodes with considering the non-hierarchy and the existence of loops. Each node has a table of the IP addresses of its known members in that community. Each community member doesn't need to know all the community members. For simplicity let us fix the number of members each member can know. For example in figure 1, we assume each member knows four members at maximum such as the member in node B has a table of his known member's IP addresses. Each node keeps a short memory of the recently routed messages in order to avoid re-sending. Each node has a mini search engine for searching in its local file system for the required file name to share. When a node sends a share request message for a specified file name, the members who received these requests use their own search engine for finding the requested file.

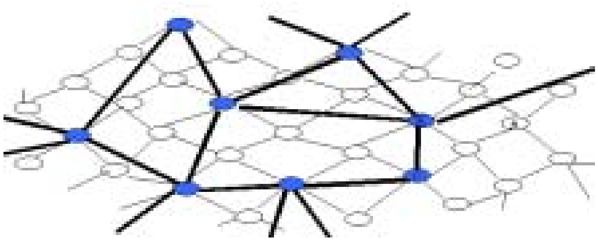

Figure 1

4.2. Communication Technology

Internet traffic, typically through Web browsers, has built upon the idea of one-to-one communication. It is not recommended to use it as community communication as a result of:

- Each member should know all the other members. He sends a message to the others one by one.
- This protocol gobbles up the network bandwidth,

Assume the communication cost from the requester to each member is fixed and equally to CT seconds. Hence the worst transmission time of a message is

$$T_{one\text{-}one} = M * CT$$

Where, the searching time at each node is neglected, and M is the number of members.

While our proposed community communication technique performs the communication among the community members based on a one-to-many (1->N). In this communication method, each member doesn't need to know all the community members but he needs only to know a specified number of members (neighbors). The community member sends N messages to N members. Then, each member from those N members re-sends the message to another N members and so on gradually level by level. Community communication technology handles under the model known like **viral propagation**. One-many (1->N) communicat-ion assures that at maximum M members received the broadcasted message.

$$M = \sum_{i=1}^{TTL} N^{i}$$

To avoids messages from being re-send indefinitely through the network by attaching each message by a flag so called **time to live** (TTL). The number of layers TTL can be calculated as the integer value from the following equation.

$$\left(\frac{\log \left[(N - 1) M + 1 \right]}{\log N} \right) - 1 \leq TTL$$

The community member should carefully scrutinize the TTL and minimize as necessary. Mishandle of the TTL will lead to an unnecessary amount of network traffic and poor network performance. The worst transmission time of a message is

$$T_{one-many} = \underset{i-1}{\overset{N_P}{Max}} \; T_i$$

The searching time at each node is neglected, N_P is the number of paths and T_i is the time required for a message to be arrived from the requester to the last member in this path i. It is a summation of the time required to send a message from each member to the next one in this path.

As shown in figure 2, the requester A sends a message to M=13 members. The gray nodes refer to the path having the maximum time.

We can compare the worst transmission time in one-one communications with one to many. As shown in figure 3, It is clear that using one-many (1->N) has small time than one-one while N increases the transmission time decreases and the number of levels decreases until a point, then by increasing N it is useless because the transmission time increase.

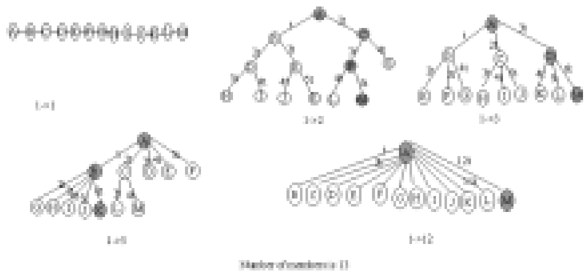

Figure 2.

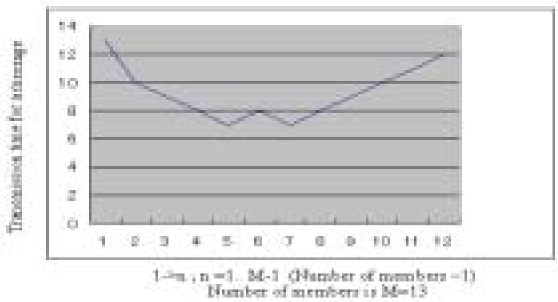

Figure 3.

The transmission time of a message based on one-many is less than the transmission time of a message based on one-one (i.e. $T_{one-one} > T_{one-many}$). Our community communication technology is based on one-many communication protocol. Figure 3, represents the effectiveness of using one-many communication protocol in comparing with one-one-communication protocol. In case of one-many (1->N), M is the number of members, and N is equal to M-1 then the worst transmission time of a message becomes equal to the transmission time of the same message using one-one communication. In this case the requester knows all the community members in which a serious problem will be raised if the number of the community members is becoming very large. Moreover it is contradicted with the assumption that each member knows only few members and doesn't know all the community members.

4.3. Communication Message Formats.

We propose message format for communication among the community members. The required messages for community communication are share request message and reply share message. These message formats are used for a communication among the members. Our communication protocol avoids messages from being re-broadcasted indefinitely through the network by attaching each message by a flag TTL. The message format as follows:

i. Share request message contains the content code, share requested file name (or requested service), time to live (TTL) and the number of hops (HOPS).

Content code	File Name	TTL	HOPS

ii. Reply message contains the content code, shared requested file data (or service reply), and the number of hops (HOPS).

Content code	File Data	HOPS

These messages are utilized by the autonomous community communication techniques. In the next subsections we will describe these techniques.

4.4. Sharing Technique.

A community member creates and initiates a broadcast of a message as well as re-broadcasts others (receiving and transmitting to neighbors). For a member (say A) who acquires to download a specified file name, we propose an efficient sharing technique with the following scenario:

1. Member A sends a request to his own node. When a node receives a request, it first checks its own database for the data and returns it if found, together with specifying its source of the data. If not found, it will go to the next step.
2. Member A sends a share request message to all his known members (For example, B and C). (i.e. one-to-two) with TTL=6. Figure 4 shows the logical relation among the members and the flow of a share request message. The originator of this request is a member A. The doted arrows signify the messages, which are rejected for avoiding the replication of the same requests.

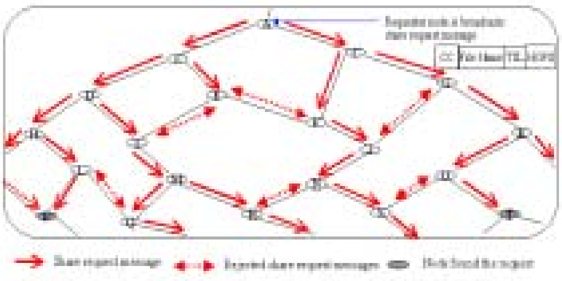

Figure 4.

3. Each member uses his own mini search engine for matches the request of the member A. If he doesn't find any, then he doesn't reply for avoiding the member A from being hailed with no results messages. Moreover, he re-sends the share request hop by hop. At each hop the TTL is decremented and the HOPS is incremented. The message is plummeted, as soon as a node observes a message with a TTL of zero. Figure 4 presents the sharing request message flow.
4. In case of one node say D as in figure 4, received a share request message while it broadcasted the same request before then D doesn't broadcast the share request message again in order to avoid the network congestion with the same request multiple of times.
5. If there are one or more nodes (P and T) match the sharing request, then they will send a reply message containing the requested file data (or service reply) hope by

hope to the node A. The reply method as follows.

4.4.1. Reply by Broadcasting

As shown in figure 5, reply node broadcasts a reply message to its neighbor nodes (i.e. one-many). At each hop the reply request message is broadcasted and the HOPS is decremented. The reply message is plummeted as soon as a node observes a message with a HOPS equal zero. Of course this method consumes more messages while it assures that the other members who interest by this data file (or service reply) can get it more easily (up to door service). This reply method assures that the reply message can be arrived to the requester in case of any intermediate node (say node D in figure 5) in the reply path is failed or the member leave the community. Wherever there are another paths may still have no failure. In the next section, we will introduce our reliability model in which we prove that our communication technique is reliable

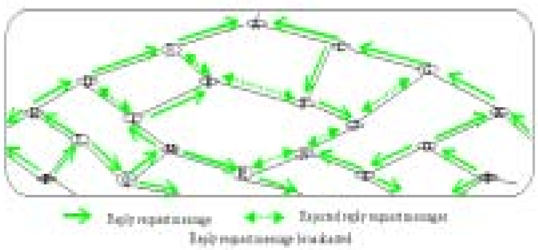

Figure 5.

5. Reliability Model

Reliability can be defined as the probability of a community node to receive a correct information without error over a given instance time t. Each community node has a number of redundant paths, which is determined by the node connectivity N. Each path I has failure rate λ_i and L_i is the number of hops from the source community node that provide the information to the receiver one in path I. The probability of path I to work probably without failure over time t defined by the exponential failure law $\alpha_i(t) = e^{-\lambda_i t}$, [8]. Moreover, the communication cost between any two-community nodes is C and the information transfer between any two-community nodes is corrupted by fixed rate τ. This model takes into account data corruption in information transformation and path failure in which may affect in receiving correct information at certain node. Path I contains nodes S_0, S_1, .., S_k whereas S_0 is the source of information as shown in figure 6-a. The reliability of node S_k to receive correct information from node S_0 through path I at time t was $\left(\alpha_{k-1}(t)\,(1-\tau)^k\right)$. The Reliability of receiving correct information at node X (see figure 6-b) at time t is

$$R_x(t) = 1 - \prod_{\substack{i=1 \\ c * L_i \le t}}^{N} \left[1 - \alpha_i(t)\,(1 - \tau)^{L_i}\right]$$

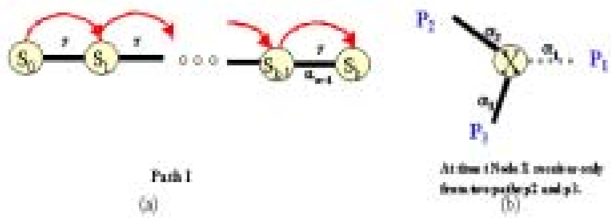

Figure 6

Each community node receives the same information from multiple paths and judges autonomously which path is reliable and attains this information from this path. Each node should wait until receive the information from all paths. Each node (say X) should wait. The waiting time (**WT**$_x$) must not be exceeds than $\left(C * \underset{i=1}{\overset{N_x}{Max}} \; L_i \right)$; where N_x is the connectivity of node X. The increasing of the nodes connectivity tend to increase the reliability as a result that each node selects the most reliable path but at the same time the waiting time at each node increased and the transmission time increased too. The question here is what the optimal waiting time at each node to avoid the increasing of the transmission time. We propose a method in which a community member sends his request accompany with a new field. It represents the importance level of the request. The importance level of the community member request varies from one to one. For example it could be a real time field (deadline) in which the community member needs to attain his request during an interval of time. This interval of time could be the allowable access time of the community member to access the Internet. Figure 7 shows that increasing of the waiting time tends to increase of the reliability and simultaneously the probability of the requester for receiving this information before deadline decreases. The trade-off point in figure 7 represents the optimal waiting time.

We conclude that our proposed communication technique among members is reliable and productive.

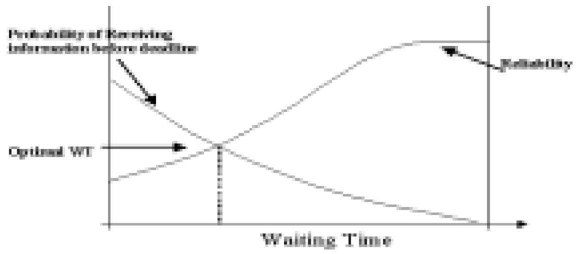

Figure 7.

6. Conclusion

We proposed the autonomous community concept in which the community members should be cooperative to meliorate their lives. The community satisfies the objectives of dynamic construction, fault tolerance and productive cooperative among its members. In addition to we proposed an efficient community communication technology, which based on one-many communication protocol. This communication protocol is reliable and productive. Our proposed autonomous community serves the Internet users efficiently with low efforts and retains the network bandwidth.

References

[1] K. Mori, Autonomous Decentralized Systems: Concept, Data Field Architecture and Future Trends, Proc. of the first Int. Sym. on ADS, (ISADS'93), IEEE, Kawasaki, Japan, pp. 28-34, 1993.

[2] K. Mori, H. Ihara, et al., Autonomous Decentralized Software Structure and its Application, Proc. IEEE FJCC'86, pp. 1056-1063. November 1986.

[3] K. Ragab, T Ono, N. Kaji, and K. Mori, Autonomous Community: Concept and Communication Techniques, submitted to ISCC'02 Italy.

[4] Napster, homepage http://www.napster.com

[5] Scour Exchange, http://www. scour.com

[6] F. Bordignon and G. Tolosa, Gnutella: Distributed System for Information Storage and Searching Model Description, University Nacional de Luján, Department of Basic Sciences, Argentina.

[7] H. Rheingold, The Virtual Community, http://www.rheingold.com/vc/book/3.html

[8] Neil Storey, Safety-Critical Computer System, Addison Wesely Longman 1996.

[9] The world's leading resource for Internet trend and statistics, http://www.nua.com/surveys/