A Method to Construct an Attack and Fault Tolerant Scalable Distributed Network

---

Citation:

Katsuya Suto, Hiroki Nishiyama, Hideaki Yoshino, Keisuke Ishibashi, and Nei Kato, "A Method to Construct an Attack and Fault Tolerant Scalable Distributed Network," 4th International Conference on Communications, Mobility, and Computing (CMC 2012), Guilin, China, May 2012.

A Method to Construct an Attack and Fault Tolerant Scalable Distributed Network

Katsuya Suto[1,§], Hiroki Nishiyama[1], Nei Kato[1], Kohei Shiomoto[2], Hideaki Yoshino[2], and Keisuke Ishibashi[2]
[1]Graduate School of Information Sciences, Tohoku University, Sendai, Japan
[2]NTT Service Integration Laboratories, NTT Corporation, Tokyo, Japan
E-mail: suto[§]@it.ecei.tohoku.ac.jp

***Abstract*—Distributed networks have attracted much attention due to their scalability and inexpensiveness as compared to traditional centralized networks. While distributed networks are appropriate to construct large-scale networks, insuring the tolerance to network-failures (i.e., attacks and faults) and communication efficiency is still an unresolved issue. In this paper, we classify and evaluate the existing distributed networks based on their degree distributions. We also propose a method to construct a network based on bimodal degree distribution, which is tolerant to network-failures insuring high communication efficiency. Additionally, through computer simulations, we show that the proposed method achieves higher tolerance compared to other existing methods.**

I. Introduction

Although large-scale networks are necessary to promote ubiquitous communications, centralized networks such as traditional client/server system are not appropriate to construct such a large-scale network, because they lead to increased load on servers. Therefore, distributed networks have attracted attention due to their high scalability and low expense. While nodes are classified into servers or clients in centralized networks, each node plays either the role of server or client according to the situation in the distributed network. In other words, nodes are equally connected each other in the distributed network, which is appropriate to construct large-scale network consisting of a large number of nodes and to distribute traffic load over the whole network. Peer-to-Peer (P2P) networks and grid networks are notable examples of distributed network designs. Internet telephony, file sharing, and media streaming services have been widely deployed on P2P technologies [1], and large-scale computing are built on grid computing technologies [2].

However, distributed networks suffer from low tolerance to network-failures, due to the lack of centralized equipment to manage the frequent joining and leaving of enormous number of nodes. In this paper, we consider two different kinds of network-failure causes, i.e., attacks and faults. Network-failures due to faults are unavoidable in any network, while it is difficult to completely prevent the network from failures caused by attacks. The failures cause network performance degradation, such as, low communication efficiency or communication disruption. Thus, improving the tolerance to network-failures and the communication efficiency is a significant issue to provision various kinds of services over the distributed network [3]. In this paper, we propose a method to construct an attack and fault-tolerant distributed network based bimodal degree distribution.

The rest of this paper is organized as follows. In Section II, we focus on the degree distribution of traditional distributed networks which are classified into several categories based on their degree distributions, and the features of these network are briefly described. Our proposed method is introduced in Section III, followed by Section IV with its performance evaluation, conducted through computer simulations from the view point of the network connectivity and communication efficiency. We conclude this paper in Section V.

II. Degree Distribution of Distributed Networks

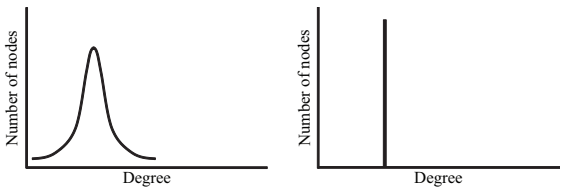
Since most of the distributed networks are constructed by joining a node into the network one-by-one, we focus on the node joining method. Different node participation ways create different networks having different degree distributions. In this section, we describe three typical examples of distributed networks, random network, regular network, and scale-free network, which have different degree distributions. Degree distribution is one of the most standard metrics which is often used for quantitative evaluation of network performances [4]. Degree distribution represents the frequency distribution of the number of links connected to each node in the network.

A. Random network

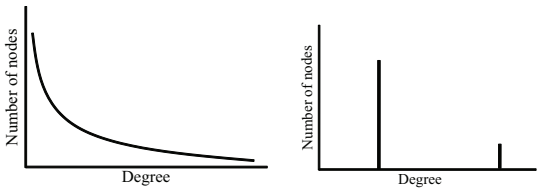
Random networks have a degree distribution following the normal distribution as shown in Fig. 1(a). Random networks are generally constructed such that a newly joining node randomly selects a certain number of nodes as its neighbors. The list of nodes within the network is provided to the newly joining nodes by the network upon joining. The random network has a narrower degree distribution and there are no high degree nodes connecting thousands of nodes, and hence the communication efficiency and the attack tolerance is low. While such drawbacks can be improved by increasing the average degree, it is not a feasible solution in practical terms due to non-negligible overhead to establish connections with a lot of nodes.

B. Regular network

In regular networks, all nodes have the same degree as depicted in Fig. 1(b). Regular networks are generally constructed such that a newly joining node selects a certain



(a) Random network (b) Regular network



(c) Scale-free network (d) Bimodal network

Fig. 1. Degree distributions of each network

number of nodes as its neighbors so as to equalize the degrees of all nodes in the network. For example, by choosing lower degree nodes their, nodes in the network will have degrees closer to the average network degree. In regular network, the communication efficiency is low if the average degree is low because of non-existence of high degree nodes in the network, while the attack tolerance is better than that of random network which has a larger variance in its degree distribution.

C. Scale-free network

As it is well known, a lot of existing networks, such as hierarchical P2P networks [5] and World Wide Web (WWW) [6] are considered as instances of scale-free networks which have power-low degree distributions as shown in Fig. 1(c). A scale-free network can be constructed by the probability nodes selection, i.e., a newly joining node stochastically chooses a certain number of nodes as its neighbors based on the probability proportional to the degree of each candidate. In scale-free networks, which are categorized by broader degree distributions, there are a few high degree nodes, which allows to achieve high communication efficiency at the price of low tolerance to attacks aiming at the high degree nodes and the traffic convergence to them. In addition, the performance of scale-free networks can be drastically degraded by the elimination of a few number of high degree nodes.

III. Bimodal Network

The communication efficiency is key metric to construct distributed networks. Because random networks and regular networks have extremely-low communication efficiency, scale-free networks is only method on the realization of distributed networks. However, the existence of high degree noes than other nodes is not preferred in terms of threat of attacks; nodes having high degrees become targets of attacks because their breakdown adversely affects the whole network. In order to achieve high robustness to both attacks and faults insuring high communication efficiency, we focus on bimodal degree distribution introduced by T. Tanizawa *et al* [7]. In the remainder of this section, we briefly describe the bimodal degree distribution, and propose a method to construct the network based on bimodal degree distribution, called bimodal network.

A. Bimodal degree distribution

As shown in Fig. 1(d), there exist only two different types of node in the bimodal degree distribution, i.e., a small number of high degree nodes, and a large number of low degree nodes. In contrast with the power-law degree distribution, the bimodal degree distribution can achieve high tolerance to attacks because the existence of only two types reduces the vulnerability against attacks. The degree of each node is defined by the following equations:

$$k_{\mathrm{low}} = \langle k \rangle, \tag{1}$$

$$k_{\mathrm{high}} = \sqrt{\langle k \rangle N}, \tag{2}$$

where $\langle k \rangle$ denotes the average degree of the network. The total number of nodes, N can be expressed as follows:

$$N = N_{\mathrm{low}} + N_{\mathrm{high}} = (1 - r)N + rN, \tag{3}$$

where N_{high} and N_{low} indicate the number of high degree nodes and low degree nodes, respectively. r represents the optimal ratio to achieve the highest attacks and fault-tolerance. The value of r is derived from the statical analysis [7] as follows:

$$r = \left(\frac{A^2}{\langle k \rangle N} \right)^{\frac{3}{4}} \tag{4}$$

$$A = \left\{ \frac{2\langle k \rangle^2 (\langle k \rangle - 1)^2}{2\langle k \rangle - 1} \right\}^{\frac{1}{3}}. \tag{5}$$

B. A method to construct bimodal network

Bimodal networks can be easily constructed by using two types of lists, namely a node list and a candidate list for high degree nodes. While the node list is used to share global information and control the degree of each node, high degree nodes are selected from the candidate list. The procedure of constructing bimodal network is composed of three phases, namely selection of high degree nodes, link setup, and link and list maintenance.

1) Selection of high degree nodes: A newly joining node gets the node list, which contains a list ID, the average degree of the network, and the profile of each node (i.e., address, node type, and the number of links). The newly joining node also receives the candidate list, which includes a list ID and the profile of the candidate (i.e., bandwidth, hardware capabilities). The node with the highest performance is registered in the candidate list. Then, the newly joining node compares the ideal and the current number of high degree nodes to decide its type. While the current number of high degree nodes can be obtained from the node list, their ideal number is calculated from the $\langle k \rangle$ and N including the newly joining node by using Eq. (4). If the ideal ratio is larger than the current ratio, the newly joining node should select a node

as high degree node. On the other hand, a node with high performance is suitable to become a high degree node. Thus, the newly joining node determines high degree node using the candidate list and registers its profile on the candidate list. Moreover, the selected high degree node updates the node list.

2) Link setup: Next, the newly joining node executes two successive procedures, insertion and expansion processes, to set up links to appropriate neighbors. In the insertion process, the newly joining node randomly selects a link in the network, and inserts itself into the link; breaking the link connecting certain two nodes and constructing new links between itself and each node. It should be noted that the connectivity of the network is maintained before and after the insertion process. After the insertion process, the newly joining node begins the expansion process in which the node repeats the construction of additional links until the degree of the node reaches the decided value or no candidate for neighbor remains in the node list. Here, nodes having less degrees than the desired value are selected as the candidate for neighbors from the node list. Since only one link is allowed to exist between any two nodes, it is possible that the newly joining node cannot have enough number of links due to the lack of candidates. In such a case, the node temporarily gives up getting links and waits for the following other node participations which creates a chance to obtain new links. After the end of expansion process, the newly joining node adds itself to the node list.

3) Link and list maintenance: In order to share the latest informations over the whole network, these lists are updated periodically using a hello message. Each node u_{sender} broadcasts a hello message, which contains IDs in both of node list and candidate list. Then, the neighbor node u_{receiver} received the message compares the IDs in the message and that in its lists. If IDs in the message is old, u_{receiver} sends the latest list to u_{sender}. If IDs in the message is new, u_{receiver} sends Acknowledgement (ACK) to u_{sender}. Thus, each node can update these lists. On the other hand, the link between neighbor nodes can be also mainteined using a hello message. When u_{sender} does not receive the ACK, u_{sender} recognizes that the link is disconnected and goes to expansion process.

IV. Performance evaluation

Ruby [8] was used to execute our experiments. We evaluate the local network connectivity and the communication efficiency of the network that some nodes are removed [9]. In all the conducted simulations, scale-free networks are used for comparison; the network is constructed by the method mentioned in Section II, and employed link and list maintenance method similar to the proposed network. In the networks, the link and list maintenance procedure is periodically invoked with an interval of time Δt, which is set to any of the following values $\{1, 10\}$. The performance of networks is evaluated in terms of tolerance for network-failures. Two different kinds of network-failure, attacks and faults, are emulated as the node leaving from network. Although nodes are left from network in descending order of node degree in the case of attacks, they are randomly left from network regardless of the degree of each node in the case of faults. In both cases, nodes are leave one by one. The average degree of each network and the number of peers are set to 3 and 10^3, respectively.

A. Local network connectivity

When some nodes are removed from a network, the network might be divided into several local network due to the loss of connections. each local network is referred to as a cluster. we evaluate the connectivity of the original global network by investigating the size of clusters generated by the node leaving.

1) Maximum cluster ratio, S: Maximum cluster ratio, S, indicating the ratio of the size of maximum cluster to the size of the network after the removal of nodes is defined as follows:

$$S = \frac{N_{\mathrm{c}}}{N_{\mathrm{rmv}}}, \quad (0 \leq S \leq 1), \tag{6}$$

N_{c} denotes the number of nodes in the maximum cluster, and N_{rmv} represents the number of peers after removal of nodes. S closer to 1 implies that the network has high local connectivity.

2) Performance comparison in S: Fig. 2 demonstrates the changes of the maximum cluster ratio, S, for different numbers of nodes removed from the network per unit time, r. The value of r is varied from 0 to 20. In both cases, the bimodal network achieves high tolerance compared with the scale-free network. It is also clear that the tolerance to network-failures degrades by increasing the interval time. In addition, The decreasing degree of tolerance in case of attacks is larger.

B. Communication efficiency

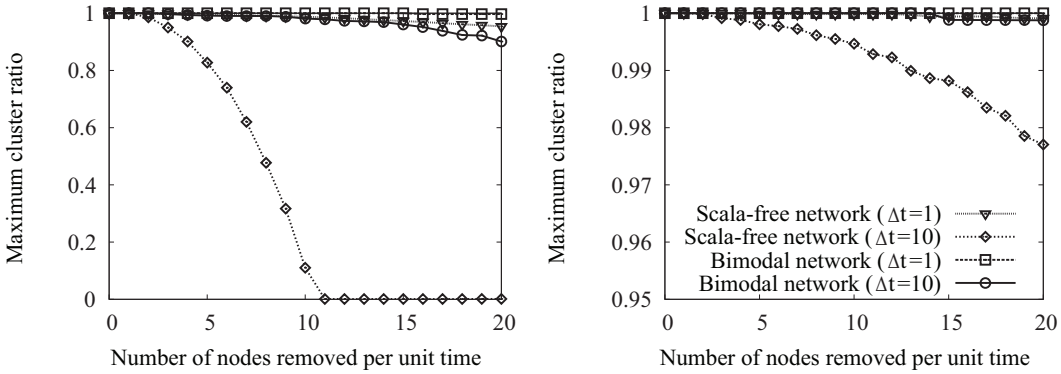
From the above discussion, we can conclude that the bimodal network are good candidates of the solution for failure-tolerant large-scale distributed network. With this as background, we turn now to communication efficiency, which is essential to evaluate performance of the distributed network. And we demonstrate that the bimodal network is the best method conclusively.

1) Metric for communication efficiency, E: To quantify the communication efficiency, we introduce the metric, E, defined by the following equation by using the inverse of the average hop counts between any two nodes in the network.

$$E = \frac{1}{N(N-1)} \sum_{i \neq j} \frac{1}{d_{ij}}, \quad (0 \leq E \leq 1), \tag{7}$$

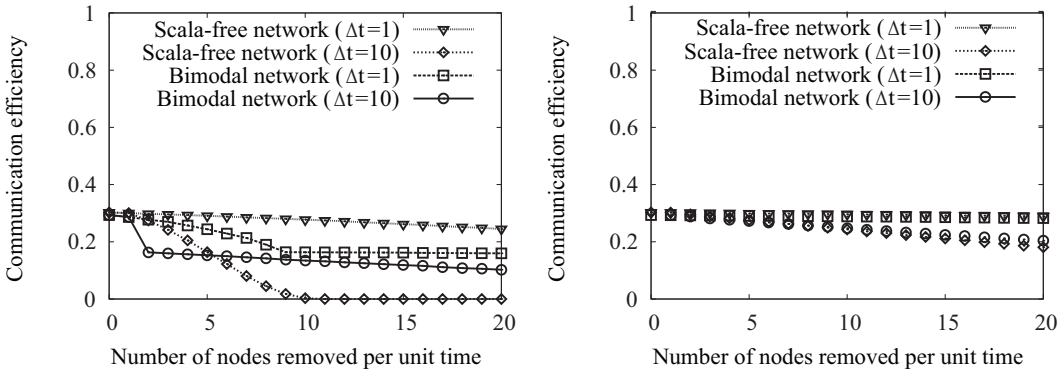
where d_{ij} denotes the number of hops between i^{th} node and j^{th} node. Here, if there is no available path between two nodes, the hop count between them corresponds to the infinity, i.e., the inverse of the hop count is equal to 0. The maximum value of E is 1 and a large value indicates high communication efficiency.

2) Performance comparison in E: Fig. 3 depicts E for different values of r. The simulation parameters are set similar to that in the evaluation of local network connectivity. While the communication efficiency of both networks is nearly equal in case of faults, the scale-free network with $\Delta t = 1$ achieves



(a) In case of attacks (b) In case of faults

Fig. 2. Relation between the number of nodes removed per unit time, r, and the maximum cluster ratio, S.



(a) In case of attacks (b) In case of faults

Fig. 3. Relation between the number of nodes removed per unit time, r, and the communication efficiency, E.

the highest communication efficiency in case of attacks. However, its communication efficiency drastically decreases as the link maintenance interval increases. This is because a few high degree nodes are removed from the network. On the other hand, the bimodal network keeps high communication efficiency irrespective of the link maintenance interval. Thus, we can conclude that the proposed bimodal network is the most suitable method for network tolerant to attacks and faults.

V. Conclusion

In this paper, we have proposed a method to construct an attack and fault-tolerant distributed network. Distributed networks can be classified according to their degree distributions into random, regular, scale-free, and bimodal network. From the fact that the bimodal network is one of the best solutions to achieve both attack and fault tolerance compared with the existing networks, we have focused on bimodal degree distributed networks. Through extensive computer simulations, we have shown that the network constructed by the proposed method can offer not only high connectivity but also high communication efficiency.

References

[1] D. Chopra, H. Schulzrinne, E. Marocco, and E. Ivov, "Peer-to-peer overlays for real-time communication: Security issues and solutions," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 4–12, 1st Quarter 2009.

[2] R. Ranjan, A. Harwood, and R. Buyya, "Peer-to-peer-based resource discovery in global grids: A tutorial," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 2, pp. 6–33, 2nd Quarter 2008.

[3] D. Chopra, H. Schulzrinne, E. Marocco, and E. Ivov, "Peer-to-peer overlays for real-time communication: security issues and solutions," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 4–12, 1st Quarter 2009.

[4] R. Albert, H. Jeong, and A.-L. Barabasi, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, Jul. 2000.

[5] M. Ripeanu, A. Iamnitchi, and I. Foster, "Mapping the gnutella network," *IEEE Internet Computing*, vol. 6, no. 1, pp. 50–57, Jan.-Feb. 2002.

[6] R. Albert, H. Jeong, and A.-L. Barabasi, "Error and attack tolerance of complex networks," *Nature*, vol. 401, pp. 130–131, Sep. 1999.

[7] T. Tanizawa, G. Paul, R. Cohen, S. Havlin, and H. E. Stanley, "Optimization of network robustness to waves of targeted and random attacks," *Physical Review E*, vol. 71, no. 4, p. 047101, Apr. 2005.

[8] "Ruby [Online]." Available: http://www.ruby-lang.org/en/.

[9] S. Sun, Z. Liu, Z. Chen, and Z. Yuan, "Error and attack tolerance of evolving networks with local preferential attachment," *Physica A: Statistical and Theoretical Physics*, vol. 373, no. 1, pp. 851–860, Jan. 2007.