Journal of the Korean Physical Society, Vol. 44, No. 3, March 2004, pp. 628∼632

Response Network Emerging from Simple Perturbation

S.-W. Son,* D.-H. Kim, Y.-Y. Ahn and H. Jeong†
Department of Physics, Korea Advanced Institute of Science and Technology, Daejeon 305-701

(Received 8 October 2003)

In complex networks, an attack on a single node can drastically change the communication pattern between other nodes. To investigate this effect, we measure the betweenness centrality (BC) changes after single node removal. Then we construct a corresponding secondary network based on this response characteristic of the network under perturbation. We find that the changes of BC are proportional to the BC of a removed node. We use a minimal spanning tree and a percolation cluster method for network construction and find that the degree distribution of secondary networks follows a power-law distribution in both cases.

PACS numbers: 05
Keywords: Scale-free network, Betweenness centrality

I. INTRODUCTION

In 1959, Hungarian mathematician Paul Erdös and his collaborator Rényi proposed a random graph model to describe several networks in nature. This was an outstanding achievement in graph theory and served as a standard model of various networks for a while. By the 1990s, the performance of computers had become more efficient than in the 1960s and scientists started to examine complicated networks more precisely with this computing power, and found many interesting features of complex networks, such as degree distribution, clustering coefficient, centrality, *etc.* They realized that the degree distribution of a real network was not the same as that of a random network. It turns out that the degree distribution of some real networks follows a power-law distribution, not a Poisson distribution as expected from the Erdös and Rényi model. To explain this discrepancy, in 1999 a new model was proposed by Barabási and Albert [1], called the scale-free (SF) model because it has a correct power-law degree distribution. After that, it was reported that various real-world networks - Internet, WWW, e-mail network [2,3], collaboration network [4–7], protein network, metabolic network [8], *etc.* - also had the same scale-free structure. (For more information, see [9,10])

In biology, the microarray technique enables us to collect large amounts of genetic data at once. Experimental data from a microarray contain the automated result of each gene's response to specific experiment. More precisely, microarray data contain each gene's relative change in its expression level under specific experimental conditions. Common experiments include changes in temperature, pH and, most importantly, single gene knock-out. We all know that the genes are correlated in the sense that some of them are closely related through some biological network (such as gene regulation) and some are not. From these microarray data, we can define the correlation between two genes by measuring similarity/dissimilarity of their expression level during the experiment. Based on these correlations, we can build a gene-gene interaction network by connecting most correlated genes together, which might represent the underlying biological network of a specific organism. This network might not be a real/physical network; however, it certainly has a biological significance. Like building this genetic network from microarray data, the response of the original network under perturbation can be used to construct a secondary network which may be different from the original network. The perturbation, for example single node deletion from the network, can be understood as a gene deletion experiment in the microarray data. In this sense, we can easily imagine that the secondary network will represent the primary network as the genetic network represents the biological network.

However, the dynamic response of the scale-free network under perturbation has not yet been investigated extensively. Albert *et al.* showed in their paper that a scale-free network is more tolerant toward random removal of nodes than a random network [11]. Recently, Kim *et al.* measured the characteristic path length (CPL) - average distance between two nodes in the network - changes in a scale-free network under a node removal and found that they followed a power-law distribution [12]. These studies suggest that the scale-free network has its own characteristic structure, which shows non-trivial behavior, quite different from the ran-

*E-mail: sonswoo@kaist.ac.kr
†E-mail: hjeong@kaist.ac.kr

dom network when there is a perturbation.

Most networks are not static objects, but dynamic objects in the sense that they grow in time, and even change their structure as well. Even though investigation of this evolution of a network structure is a challenging and interesting problem, in this paper we will consider the simplest dynamic procedure of structural changes of a network, a node removal. In 1977 the sociologist Freeman defined the quantity called betweenness centrality (BC) [13] to measure the importance of a node in network communication. BC of a node represents the average amount of traffic passing through that node. Precisely, the node BC is defined as

$$b(k) = \sum_{i,j} b_{i\to j}(k) = \sum_{i,j} \frac{g^k_{i\to j}}{g_{i\to j}}. \quad (1)$$

In Eq. (1), $g_{i\to j}$ is the number of geodesic paths from i to j and $g^k_{i\to j}$ is the number of paths from i to j that pass through k. In this definition, we assume that all pairs of nodes communicate once with each other through the shortest geodesic path. This is not always possible because, to find the shortest path, we have to know the whole network structure in advance. However, it was found that this assumption was quite reasonable up to some practical level [14]. If we imagine everyday communication like a phone call over a phone line or sending/receiving email via Internet, it is obvious that node BC corresponds to the traffic burdens or loads which must be endured by a node on the network to maintain the communication between nodes [15]. The most important property of BC that is quite different from the degree is that it carries global information on the network. The degree contains the connectivity information of the node with its neighborhood, which is local information. On the other hand, the BC contains global information, because we calculate the BC of a node by summing the local information over the whole network. Therefore, the BC has been considered as a good measure for scale-free network classification [16].

II. METHOD

The BC is a computationally involved network measure. The original algorithm computing BC based on definition Eq. (1) can deal with networks of up to several hundred nodes. However, with the increasing availability of electronic data collection and, of course, the advent of the web, there is an increasing demand for computation of BC on networks with a large number of nodes. As a remedy, a new algorithm for betweenness has been proposed by Ulrik Brandes [17, 18]. It exploits the extreme sparseness typical of real networks; it runs considerably faster and consumes much less memory. In technical terms, the running time is reduced from $O(MN^2)$ to $O(N^2 + NM)$ and memory consumption is reduced from $O(N^2)$ to $O(N+M)$, where N is the number of nodes and M the number of edges. Moreover, this algorithm can compute other shortest-path-based measures, such as closeness, simultaneously within the same bounds. This algorithm significantly extends the size of networks for which betweenness can be calculated.

First, we calculate the BC of all nodes in the network and store these results in memory. We denote this original BC of node k as $b^{(0)}(k)$. Then, we choose one node i from the network and remove this node along with all edges that are connected to node i. After removing node i, we again calculate the BC of all remaining nodes. We denote $b_i(k)$ for the BC of node k after node i removal and $\Delta b_i(k)$ for the BC difference before and after node removal. Because we are only interested in single node removal, we restored the removed node and repeated this procedure for each node in the network.

$$\Delta b_i(k) = b_i(k) - b^{(0)}(k). \quad (2)$$

Even with a brute force algorithm on a single PC, one calculation of BC for ten thousand nodes requires only a few minutes, but a whole set of calculations with every node removed consumes at least ten thousand minutes, about one week. Therefore, an efficient algorithm and great computing power are essential for this calculation. As stated above, we used Ulrik Brandes' algorithm which has complexity $O(MN)$ for one calculation of the BC of an unweighted network. Therefore, the complexity of the required calculation for betweenness centrality difference is reduced to $O(MN^2)$, which corresponds to only a few minutes.

After we calculate $\Delta b_i(k)$ for every node k with node i removal, we construct a matrix $\Delta b = (b_{ij})$ (see Eq. (3)) from the results:

$$\mathbf{\Delta b} = \begin{bmatrix} \Delta b_1(1) & \cdots & \Delta b_1(j) & \cdots \\ \vdots & \ddots & \vdots & \\ \Delta b_i(1) & \cdots & \Delta b_i(j) & \\ \vdots & & & \ddots \end{bmatrix}. \quad (3)$$

The matrix dimension is $N \times N$. It looks like the adjacency matrix of a weighted graph with N nodes, edges of which are connected to every node in the network with corresponding weight b_{ij}. The weight b_{ij} represents how two nodes i and j are related indirectly. That is, how much influence has node j had on removal of node i, which corresponds to the gene expression level change in a microarray data matrix. From this adjacency matrix, we can build the secondary network. There are many ways to build a connected graph from an adjacency matrix. In this paper, we used two methods to build the secondary network; these are the minimum spanning tree and the percolation approaches.

The minimum spanning tree (MST) is a widely used method to find an optimized solution in graph theory or data structure in computer science. Prim (1957) and Kruskal (1956) developed polynomial time algorithms to

find MST [19, 20]. For a given weighted network, the MST is uniquely determined with $(N-1)$ edges, where N is the number of nodes. The MST is the spanning cluster with the least weight sum. Because it is reasonable to connect two nodes with higher correlation (Δb_{ij}), in our simulation we choose to connect the edge with largest weight first, to find the substructure which represents the maximum influential network. We put an edge between a node and its most influential node, with a constraint that every set of N nodes must be connected with only $(N-1)$ edges.

The percolation method is easier to implement than MST. While the MST method has a constraint that the tree cannot make a loop and must be connected with only $(N-1)$ edges, the percolation method does not require such constraints and can have more than $(N-1)$ edges. After sorting all edges with $\Delta b_i(j)$ in descending order, we add an edge between nodes i and j following this order. When all nodes are connected and become a single giant cluster, we stop the attachment of edges. In other words, this means that the edges with weight $\Delta b_i(j) > b^*$ are identified as valid edges and can be used to connect two nodes i and j. In this sense, we can call b^* the percolation threshold because, if we consider this process as a percolation process, connecting all bonds with $\Delta b_i(j) > b^*$ guarantees that the whole system is connected; this means the system is percolated. The percolation approach also has the important meaning that a node is connected to the highly correlated nodes first, in a similar way to the MST method. Due to the limitation of MST (*i.e.* the total number of edges should be $N-1$), some information can be lost in the MST method; however, the percolation approach can make good this missing part.

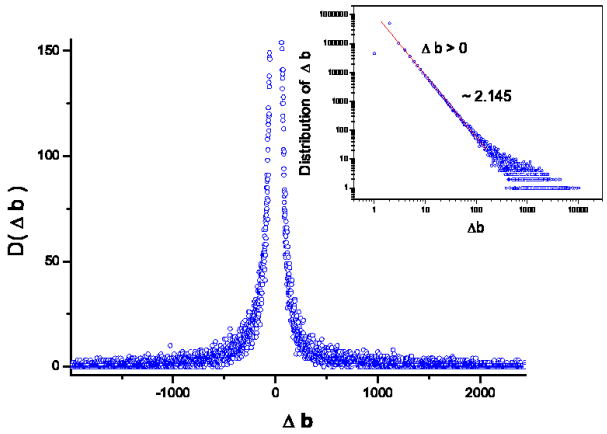

Fig. 1. Distribution of BC changes. Inset shows the distribution of the positive part in a log-log plot.

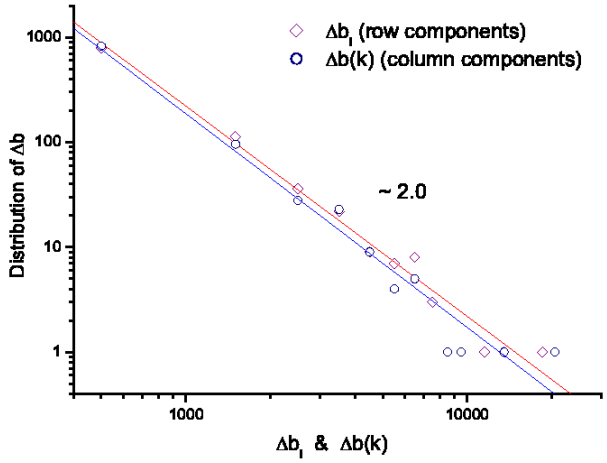

Fig. 2. Distribution of $\langle \Delta b_i \rangle$ and $\langle \Delta b(k) \rangle$, which are the average of BC changes over rows ($\langle \Delta b_i \rangle$) and the average over columns ($\langle \Delta b(k) \rangle$), respectively.

III. RESULTS AND DISCUSSIONS

1. Statistical Properties

For the Barabási-Albert (BA) model [1] with 1000 nodes and 1996 edges, we calculate the BC changes under a node deletion $\mathbf{\Delta b}$. We find that the distribution of BC changes, $\Delta b_i(k)$, follows the power law (see Fig. 1). Because $\Delta b_i(k)$ can be negative or positive, we investigate only positive values to find out the power-law distribution of BC changes. Even if we choose absolute values instead of positive values, the power-law distribution is not changed significantly.

It is more meaningful to study the distribution of column and row averages of $\mathbf{\Delta b}$ matrix elements, because the column and row averages correspond to the average of perturbed BC of each node under an arbitrary node removal and the average perturbed BC over all nodes under a node removal, respectively. Interestingly, the column and row averages of $\mathbf{\Delta b}$ also follow the power law, though the exponent is slightly different (see Fig. 2).

The power-law distribution of average BC changes over either columns or rows is understood by reference to the strong correlation between unperturbed BC $b^{(0)}(i)$ and BC change summation over all nodes Δb_i. Fig. 3 indicates nearly linear correspondence between $b^{(0)}(i)$ and Δb_i. Because $b^{(0)}(i)$ follows the power law, Δb_i must follow the power law as well. This is quite intuitive, because it is expected that the BC change will become larger if we remove the node with larger BC.

However, the power-law exponent of Δb_i, 2.0, is different from that of unperturbed BC, 2.2, for the BA model. We find that the CPL change caused by the node removal is responsible for the change in the exponent. We derive the BC change summation Δb_i. Using the fact that the summation of BC corresponds to the CPL of the network, Δb_i can be rewritten as follows:

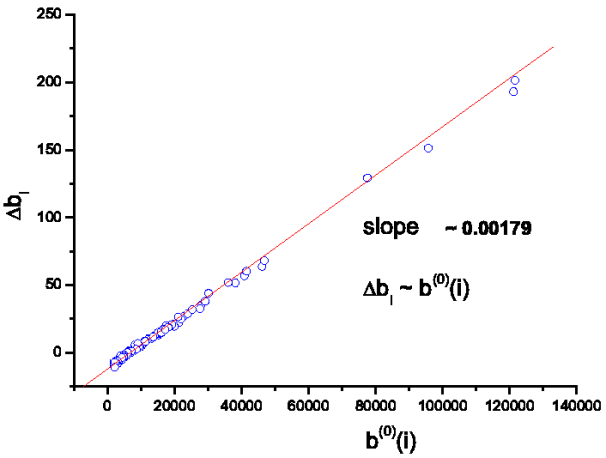

Fig. 3. Relation of the unperturbed BC and the average of the BC changes. The BC changes of each node are directly proportional to the original BC.

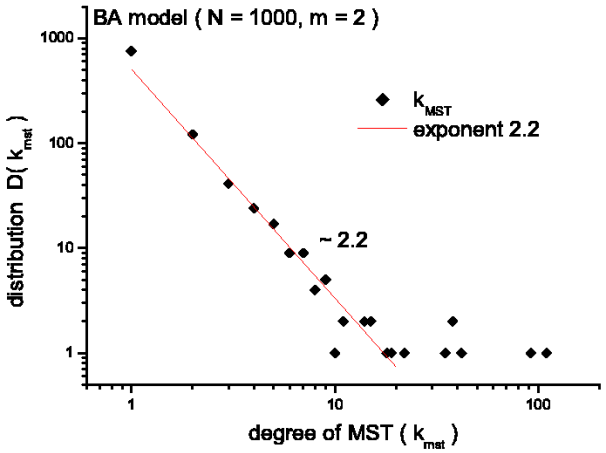

Fig. 4. Degree distribution of the secondary network constructed by using the MST method.

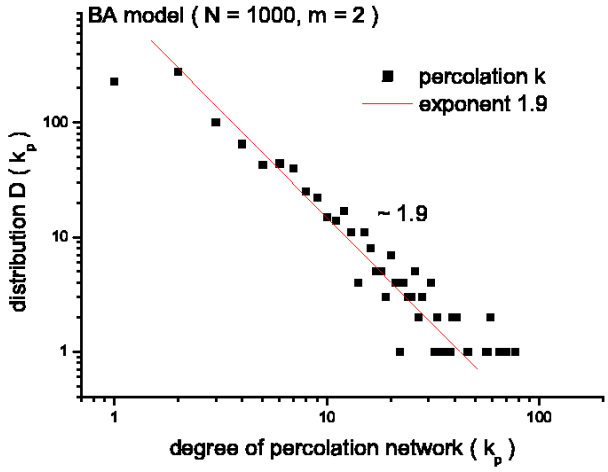

Fig. 5. Degree distribution of the secondary network constructed by using the percolation method.

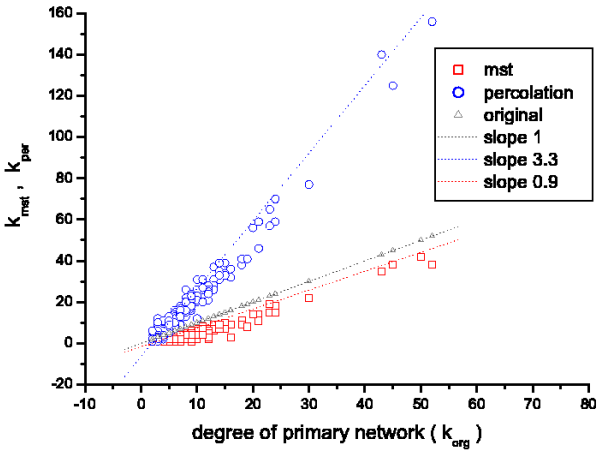

Fig. 6. Relation between degrees of the original network and the secondary network.

$$\begin{aligned}\Delta b_i &= \sum_{k\neq i}\Delta b_i(k) \\ &= \sum_k b_i(k) - \sum_k b^{(0)}(k) + b^{(0)}(i) \\ &\simeq \Delta D + b^{(0)}(i), \end{aligned} \tag{4}$$

where ΔD is the summation of the CPL change of the network. Thus, the BC change is determined by both the CPL change and the unperturbed BC, which leads to the different exponent of the BC change distribution compared to that of the unperturbed BC distribution.

2. Secondary Networks

We construct secondary networks that reflect the relation between nodes under BC changes. For instance, if $b_i(k)$ is larger, the nodes i and k are more likely to be connected. As we describe in Sec. II. , we construct the secondary networks in two different ways: MST and percolation. In the degree distribution of secondary networks, we find that secondary networks also show *scale-free* behavior (see Figs. 4 and 5). However, the exponents are very different from the degree exponent of the original network. The network constructed by using MST shows the exponent of 2.2, and that from percolation indicates the exponent 1.9, which is far from 3.0, the degree exponent of the original BA network. The degree exponent of secondary networks is similar to the exponent of BC distribution of the BA model. We guess that it might be related to the fact that BC changes governing secondary networks are nearly proportional to the BC of the original networks.

We find that the resulting secondary networks have structural similarity to the original network on comparing local properties of secondary networks, such as degree

and nearest neighbors, to those of the original network. The degrees of those networks show strong correlation in Fig. 6.

IV. CONCLUSIONS

We study BC changes under a node deletion in the BA model. We find that BC changes follow the power-law distribution, and the secondary networks constructed by using MST and a percolation method have similar local structure to their original networks. Strong correlation between unperturbed BC and BC changes of a node gives rise to the power-law distribution of BC changes. Local similarity of secondary networks and original networks indicates that the deletion of a node greatly affects BCs of nearest neighbors.

ACKNOWLEDGMENTS

We would like to thank Korea Advanced Institute of Science and Technology for generous help. This work was supported by grant No. R14-2002-059-01002-0 from the KOSEF-ABRL program.

REFERENCES

[1] A.-L. Barabási and R. Albert, Science **286**, 509 (1999).
[2] R. Albert, H. Jeong and A.-L. Barabási, Nature (London) **401**, 130 (1999).
[3] H. Ebel, L.-I. Mielsch and S. Bornholdt, Phys. Rev. E **66**, 035103 (2002).
[4] M. E. J. Newman, Phys. Rev. E **64**, 016131 (2001).
[5] M. E. J. Newman, Proc. Natl. Acad. Sci. **98**, 404 (2001).
[6] H.-J. Kim, I.-M. Kim, Y. Lee and B. Kahng, J. Korean Phys. Soc. **40**, 1105 (2002).
[7] Y. Kumagai, J. Korean Phys. Soc. **40**, 1100 (2002).
[8] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai and A.-L. Barabási, Nature (London) **407**, 651 (2000).
[9] R. Albert and A.-L. Barabási, Rev. Mod. Phys. **74**, 47 (2002).
[10] S.N. Dorogovtsev and J.F.F. Mendes, Adv. Phys. **51**, 1079 (2002).
[11] R. Albert, H. Jeong and A.-L. Barabási, Nature **406**, 378 (2000).
[12] J.-H. Kim, K.-I. Goh, B. Kahng and D. Kim, Phys. Rev. Lett. **91**, 058701 (2003).
[13] L.C. Freeman, Sociometry **40**, 35 (1977).
[14] B.J. Kim, C.N. Yoon, S.K. Han and H. Jeong, Phys. Rev. E **65**, 027103 (2002).
[15] K.-I. Goh, B. Kahng and D. Kim, Physica A **318**, 72 (2003).
[16] K.-I. Goh, E. Oh, H. Jeong, B. Kahng and D. Kim, Proc. Natl. Acad. Sci. **99**, 12583-12588 (2002).
[17] U. Brandes, J. Math. Sociol. **25**, 163 (2001).
[18] M. E. J. Newman, Phys. Rev. E **64**, 016132 (2001).
[19] R. C. Prim, Bell System Tech. J. **36**, 1389 (1957).
[20] J. B. Kruskal, Proc. Amer. Math. Soc. **7**, 48 (1956).