Vol.2009-AL-125 No.4
2009/7/21

スターグラフに基づく対費用効果に優れたP2Pオーバーレイの提案

藤田　聡[†1]

In this paper, we propose a new network topology for P2P overlay. The proposed topology is a contracted graph of an n-star, which realizes a short diameter with a small degree compared with conventional hypercubic networks such as Chord and skip graph.

1. Introduction

Peer-to-peer (P2P) networks have attracted considerable attentions in recent years. A P2P is a distributed system consisting of several nodes called peers, connected with a logical network called P2P overlay. Existing P2P networks can be classified into two categories by the way of controlling the topology of the P2P overlay and the way of data management in the network; i.e., it is either structured or unstructured. A typical unstructured P2P is Gnutella[6] proposed in 2000. In this system, the topology of the overlay is not explicitly controlled by the participating peers, and the search of a target file held by a peer is conducted through the flooding of a query message to the peers within a predetermined region centered at the requesting peer (it is generally controlled by setting TTL (Time-to-Live) to each query message). On the other hand, in structured P2Ps, we can *design* the topology of the P2P overlay and the way of data allocation, in such a way that the location of target file is quickly identified, the load of peers is balanced, and it is adaptive to the dynamic change of the set of participating peers due to join and leave of those peers.

In this paper, we propose a new network topology for structured P2P overlay. The proposed topology is a generalization of the star graph, which is known to accommodate $n!$ vertices while keeping the degree of each vertex to $n-1$ and the diameter of the network to $\lceil 3(n-1)/2 \rceil$[1]. We extend the definition of star graph in such a way that:

1) It accommodates *any* number of vertices (we use symbol N to denote the number of vertices in a graph),
2) The degree of each vertex is bounded by $\frac{2\log N}{\log\log N}(1+o(1))$, and
3) The diameter of the resultant network is bounded by $\frac{3\log N}{2\log\log N}(1+o(1))$.

Note that it asymptotically beats the performance of conventional hypercubic P2P overlays such as Chord[11] and skip graph[2] in which the degree and the diameter of the graph are both bounded as $\Theta(\log N)$. In the literature, there are several approaches to extend the definition of the star graph in such a way that it can accommodate any number of vertices[4),5),8),10]. A basic technique used in such schemes is to prepare several star graphs of various sizes (e.g., $n!$, $(n-1)!$, $(n-2)!$, and so on), and to "connect" them so as to make the number of vertices in the resultant graph to be N. For example, we can obtain a graph consisting of 30 vertices by connecting a star graph with 24 (= 4!) vertices and another star graph with 6 (= 3!) vertices through parallel edges. A similar idea has been applied to other classes of network topologies, such as hypercube[3),9),12)–15] and Kautz digraph[7].

In contract to such previous schemes, in our scheme, we will take an approach of "split and merge" of the vertices of a given network to several (sub)vertices in such a way that the total number of vertices in the network becomes N. More concretely, 1) we focus on a *contracted graph* of a star graph defined by prefixes of vertices in the star graph, and apply a marking procedure to have a set of vertices which should be split into several (sub)vertices, and 2) focus on another contracted graph which is obtained by contracting pairs of adjacent vertices in a given graph.

The remainder of this paper is organized as follows. Section 2 describes necessary notations. A naive extension of the star graph is introduced in Section 3. The proposed scheme is given in Section 4. Finally, Section 5 concludes the paper with future problems.

†1 Department of Information Engineering, Graduate School of Engineering, Hiroshima University

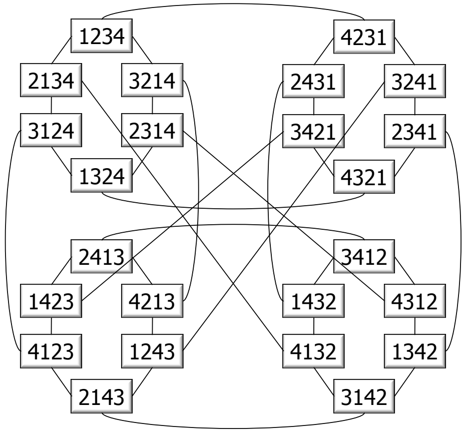


Fig. 1 Star graph S_4.

2. Preliminaries

Let V_n be the set of $n!$ permutations of symbols $\{1, 2, \ldots, n\}$. Let u_i denote the i^{th} digit of permutation u. A **generator** g_i $(2 \leq i \leq n)$ is a function from V_n to V_n which interchanges symbol u_i with symbol u_1; i.e., for given permutation $u = u_1 u_2 \ldots u_n$ $(\in V_n)$,

$$g_i(u) = \underline{u_i} u_2 \ldots u_{i-1} \underline{u_1} u_{i+1} \ldots u_n.$$

A **star graph** on n symbols (or n-star), denoted by S_n, is an undirected graph with vertex set V_n and an edge set E_n, where $E_n = \{\{u, g_i(u)\} \mid u \in V_n, \ \ 2 \leq i \leq n\}$. Figure 1 illustrates the star graph on four symbols.

In the following, we use symbol N to denote the number of vertices in a given graph. It is known that the degree of vertices in S_n is $n-1$, and the diameter of S_n is $\left\lfloor \frac{3(n-1)}{2} \right\rfloor$. Those values could be represented in term of the total number of vertices N as follows:

Remark 1 Let $N = |V_n|$. Then, the degree of vertices in S_n is at most $\frac{\log N}{\log \log N}(1 + o(1))$ and the diameter of S_n is at most $\frac{3 \log N}{2 \log \log N}(1 + o(1))$.
Proof. Since $(n/2)^n \leq n!$, we have $\log N \geq n \log(n/2) = n(\log n - 1)$. On the other hand, since $n! \leq n^n$, $\log \log N \leq \log n + \log \log n$. Thus,

$$n \leq \frac{\log N}{\log n - 1} \leq \frac{\log N}{\log \log N - \log \log n} = \frac{\log N}{\log \log N}(1 + o(1)).$$

Hence the claim follows. Q.E.D.

3. A Contracted Graph of Star Graph

Given S_n, we can construct a class of graphs by repeatedly *contracting* several vertices into a vertex. Such graph is generally referred to as a contracted graph, and our proposed scheme is based on such notion of contraction of vertices. In the following, to clarify the exposition, we introduce the notion of "bags" to represent contracted vertices. We will use symbols x and y to denote bags, and symbol B to denote a set of bags. Each bag corresponds to a prefix of a vertex in V_n. In the following, we often identify a bag with its corresponding prefix. Bag x contains *all* vertices in V_n to have prefix x. For example, when $n = 5$, bag 123 contains two vertices 12345 and 12354 in V_n.

Definition 1 Let $G(B)$ be a graph with a bag set B and an edge set E_B, where two bags x, y are connected by an edge in E_B if and only if there exist two vertices $u, v \in V_n$ such that $u \in x$, $v \in y$, and $\{u, v\} \in E_n$.
For example, two bags 123 and 423 in B are connected by an edge in E_B since there are two vertices 12345 and 42315 connected by an edge in S_5. Note that by the definition of S_n, any two vertices contained in a bag are *not* connected by an edge in E_n.

Definition 2 (Type of edges) An edge in E_B is said to be a **base edge** if it connects two bags x and y such that: 1) $x, y \in V_{n'}$ for some $n' \leq n$, and 2) x and y are connected by an edge in $S_{n'}$. The other edges in E_B are called **cross edges**.
For example, when $n = 5$, bags 123 and 213 are connected by a base edge, and bags 123 and 423 are connected by a cross edge.

For any $2 \leq i \leq n$, let $B_{n,i}$ denote a set of bags defined as follows:

$$B_{n,i} = \{u_1 u_2 \ldots u_{n-i} \mid u_1 u_2 \ldots u_n \in V_n\}.$$

For example, when $n = 5$ and $i = 3$, set $B_{5,3}$ consists of the following 20 $(= 5 \times 4)$ bags:

$$B_{5,3} = \{12, 13, 14, 15, 21, 23, 24, 25, \ldots, 54\}.$$

By definition, each bag in $B_{n,i}$ contains $i!$ vertices in V_n. For example, bag 12 in $B_{5,3}$ contains six (= 3!) vertices; i.e., $12345, 12354, 12435, 12453, 12534$, and 12543. Bag 12 is connected with bag 21 by a base edge, and is connected with three bags $32, 42$, and 52 by cross edges. Graph $G(B_{4,2})$ is illustrated in Figure 2 with a correspondance to S_4.

Remark 2 For any $2 \leq i \leq n-1$, each bag in $G(B_{n,i})$ is incident on $n-i-1$ base edges and i cross edges; i.e., the degree of each bag in $G(B)$ is $n-1$ regardless of the value of i.

The above claim indicates that by increasing the value of i, we have a sequence of graphs $G_1, G_2, \ldots, G_{n-1}$ such that: 1) $G_1 = S_n$, 2) $G_{n-1} = K_n$, and 3) $G_i = G(B_{n,i})$ for each $2 \leq i \leq n-2$. Since the number of bags in $G(B_{n,i})$ is $n!/i!$, it implies that we have a sequence of graphs consisting of the following number of bags:

$n!, n!/2, n!/3!, n!/4!, \ldots,$ and n

while keeping the degree of each bag to $n-1$. Since $n!$ is n times larger than $(n-1)!$, the above simple scheme realizes a refinement of the size of graphs, which has a big gap between $(n-1)!$ and $n!$ in the original definition of the star graph.

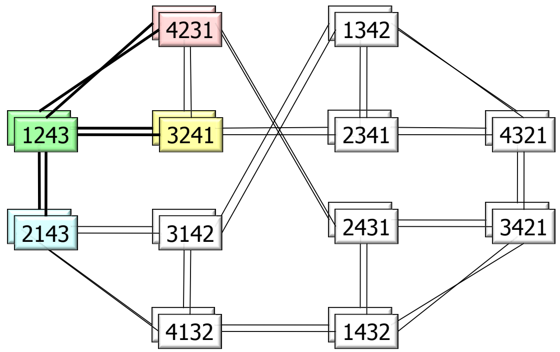
Example 1 The number of vertices in a 10-star is $10! = 3628800$ and the number of vertices in a 9-star is $9! = 362800$; i.e., the former is ten times larger than the latter. (The degree of the former one is 9, and the degree of the latter one is 8.) The above construction refines the gap, since we have a graph consisting of $10!/2 = 1814400$ vertices and another graph consisting of $10!/3! = 604800$ vertices, while keeping the degree of vertices to be 9.

4. Proposed Method

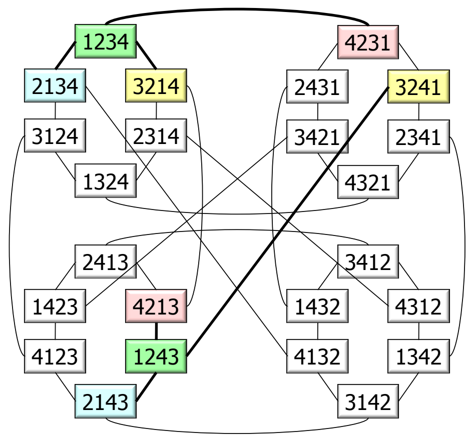
Let N be a positive integer. In this section, we propose a scheme to construct a set of bags B such that:

1) $N \leq |B| \leq 2N$,
2) each bag is adjacent with at most $\frac{\log N}{\log\log N}(1+o(1))$ bags in graph $G(B)$, and
3) the diameter of $G(B)$ is $\frac{3\log N}{2\log\log N}(1+o(1))$.

Given such graph, we can construct a graph such that: 1) it consists of N vertices, 2) the degree is at most $\frac{2\log N}{\log\log N}(1+o(1))$, and 3) the diameter is at most $\frac{3\log N}{2\log\log N}(1+o(1))$, in the following manner:



(a) Graph $G(B_{4,2})$.



(b) Correspondance to S_4.

Fig. 2 Graph $G(B_{4,2})$.

- Calculate a maximum matching of $G(B)$
- Select arbitrary $|B|-N$ edges in the matching, and contract two end vertices of each edge into a single vertex.

Those values are *asymptotically smaller* than the values for hypercubic graphs, such as Chord and skip graph.

4.1 Overview

Let n be the smallest integer such that $|V_n| \geq N$, i.e.,

$$(n-1)! \; < \; N \; \leq \; n!.$$

We use V_n as the underlying set of permutations; i.e., each digit of a string corresponding to a bag in B is drawn from set $\{1, 2, \ldots, n\}$. Let k be an integer such that

$$(k-1)! \; < \; n-1 \; \leq \; k!.$$

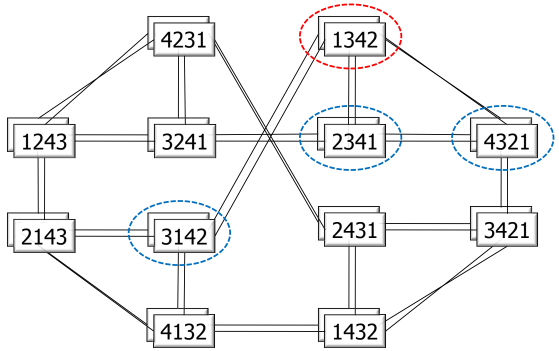
We will construct a set of bags B such that each bag in B corresponds to a prefix of a vertex in V_n of length either $n-k$ or $n-k+1$. Note that it holds $1 \leq n-k$ and $n-k+1 \leq n-1$, since $2 \leq k \leq n-1$ for any $n \geq 3$. In addition, by construction, $|B_{n,j}| > N$ for any $j < k$; i.e., $B_{n,k-1}$ contains sufficiently large number of bags even if the number of bags in $B_{n,k}$ is smaller than N.

An outline of the proposed scheme is described as follows. We start our construction from graph $G(B_{n,k})$ consisting of $n!/k!$ ($\leq N$) vertices, and will try to *mark* bags in such a way that: 1) the number of marked bags is at least $\lceil (N - |B_{n,k}|)/(k-1) \rceil$ and at most $\lceil (2N - |B_{n,k}|)/(k-1) \rceil$, and 2) each unmarked bug is adjacent with at most one marked bag. We then split each marked bag into k sub-bags, by increasing the length of the associated string from $n-k$ to $n-k+1$. See Figure 3 for illustration (in this figure, bag 13 is split into two sub-bags 132 and 134, which increases the degree of adjacent blue bags from three to four, but does not increase the degree of red sub-bags). Each sub-bag is adjacent with $n-1$ (sub)bags in the resultant graph, since the degree of bags in $G(B_{n,k-1})$ is $n-1$ as was claimed in Remark 2, and a contraction of vertices does not increase the degree of uncontracted vertices. Thus, the above splitting certainly increases the number of bags in the resultant graph, and by construction, for each unmarked bag, the number of adjacent bags increases by at most $k-1$; i.e., we can bound the degree of each bag in the resultant graph by at most $n+k-2$.

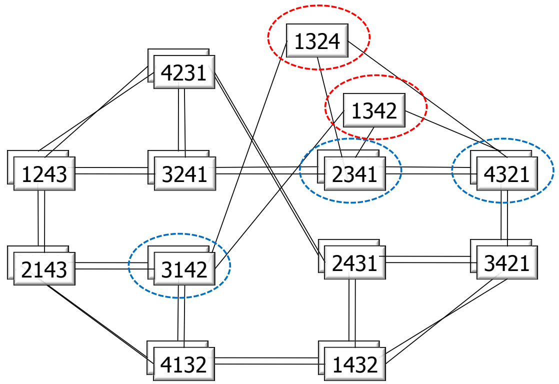
4.2 Class of Bags

Before describing the details of the proposed scheme, we introduce several notations.

Definition 3 (Class of bags) Let x be a bag of length n' ($\leq n$), and k be an integer satisfying $1 \leq k \leq n'$. Bag x is said to be of class $C_{i_1, i_2, \ldots, i_k}$, if value j appears at the i_j^{th} digit of x for any $1 \leq j \leq k$, where i_j takes value 0 if j does not appear in x.



(a) Before splitting.



(b) After splitting.

Fig. 3 Split of a bag into k (= 2) sub-bags.

In general, each bag belongs to different classes. For example, bag 123 belongs to three classes $C_1, C_{1,2}$, and $C_{1,2,3}$, and bag 234 belongs to three classes $C_0, C_{0,1}$, and $C_{0,1,2}$. Note that the number of integers in the subscript does not exceed the length of the string corresponding to the bag. In the following, to simplify the exposition, we often represent a sequence of integers as α and β (including an empty sequence); e.g., $C_{i_1,i_2,\ldots,i_j}$ is represented as C_{α,i_j} and C_{β,i_{j-1},i_j} using symbols α and β instead of sequences "$i_1, i_2, \ldots, i_{j-1}$" and "$i_1, i_2, \ldots, i_{j-2}$," respectively.

Lemma 1 Let x and y be bags belonging to classes $C_{\alpha,i}$ and $C_{\alpha,j}$, respectively, for some sequence α. If $i \neq 0$, $j \neq 0$, and $i \neq j$, then any two vertices belonging to x and y are not adjacent with each other in S_n.

Lemma 2 Let x and y be bags belonging to classes C_i and C_0, respectively. If $i \neq 0$ and $i \neq 1$, then any two vertices belonging to x and y are not adjacent with each other.

4.3 Marking Algorithm

Recall that: 1) $|B_{n,k}| = n|B_{n-1,k}|$, 2) the number of bags in class C_0 is $k|B_{n-1,k}|$, and 3) the number of bags in class C_i, $1 \leq i \leq n-k$, is $|B_{n-1,k}|$. Our proposed marking scheme is described as follows.

Algorithm

Step 1: Let $m = \lceil (N - |B_{n,k}|)/(k-1) \rceil$; i.e., m is the number of bags which should be marked by this algorithm.

Step 2: At first, we mark bags in class C_1. If $m < |B_{n-1,k}|$, then we stop the algorithm after marking *any* m bags in C_1. Otherwise, update m as $m := m - |B_{n-1,k}|$ and proceed to Step 3, after marking *all* bags in C_1.

Step 3: If $m > k|B_{n-1,k}|$, then go to Step 4 after marking all bags in classes C_0 and letting $m := m - k|B_{n-1,k}|$. Otherwise, simply go to Step 4.

Step 4: We mark bags in classes $C_2, C_3, \ldots$, and C_{n-k}, in the following manner: Let $p = \lfloor m/|B_{n-1,k}| \rfloor$ and $q = m - p \times |B_{n-1,k}|$. Note that $0 \leq p \leq n-k-1$, and $q = 0$ must hold if $p = n-k-1$. We then mark *all* bags contained in classes $C_2, C_3, \ldots, C_{p+2}$.

The correctness of the above marking scheme is certified by the following three observations:

- **Marking of C_1:** Each bag in C_i, $i \neq 0$, is adjacent with exactly one bag in C_1, and any two bags in C_1 are not adjacent. Thus, the marking of bags in class C_1 in Step 2 increases the degree of bags in C_i, $i \neq 0$, by at most $k-1$, and does not increase the degree of bags in C_1. See Figure 4 for illustration.
- **Marking of C_0:** Bags in C_0 are not adjacent with bags in C_i, for any $2 \leq i \leq n-k$. Thus, the marking of bags in class C_0 does not increase the degree of bags in C_i for $2 \leq i \leq n-k$. In addition, since the marking of C_0 is conducted only after marking of all bags in C_1, it does not increase the degree of (sub)bags generated from bags in C_1. See Figure 4 (a) for illustration.
- **Marking of other bags:** Bags in class C_i, $2 \leq i \leq n-k$, are not adjacent with bags in class C_j, for any $2 \leq j \leq n-k$ and $j \neq i$. Thus, the marking of bags in class C_i does not increase the degree of bags in C_j, if C_1 has already been marked. See Figure 4 (b) for illustration.

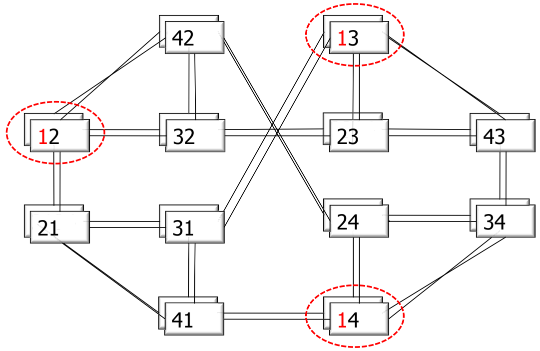


Fig. 4 Bags in class C_1.

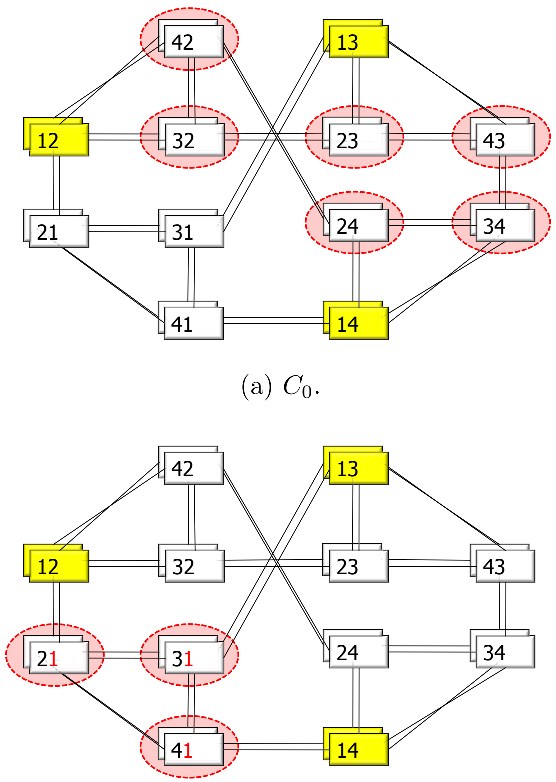
By construction, we immediately have the following claim.

Lemma 3 For any N, the proposed scheme marks vertices in $G(B_{n,k})$ in such a way that each unmarked bag in $G(B_{n,k})$ is adjacent with at most one marked bag.

4.4 Splitting

After completing the marking of bags in $G(B_{n,k})$, we split each marked bag into k sub-bags in order to align the number of resultant sub-bags to given N. Thus, we have the following theorem.

Theorem 1 The proposed scheme generates a contracted graph of S_n such that: 1) the length of strings associated with each vertex is either $n-k$ or $n-k+1$,



(a) C_0.

(b) C_2.

Fig. 5 Bags in classes C_0 and C_2.

and 2) the degree of each vertex is at most $n+k$ and at least $n-1$.

Finally, since a split of bag into sub-bags does not increase the diameter of the network, the diameter of the resultant network does not exceed the diameter of S_n, i.e., at most $3(n-1)/2$.

5. Concluding Remarks

Future problems are listed below:

- Given n and k, how to increase or decrease the number of vertices in the resultant graph, in the range of $n!/k!$ to $n!/(k-1)!$, in a distributed manner (probably we need to introduce a kind of management peer, in order to keep the current values of n and k).
- How to decrease the value of k when N increases to $n!/(k-1)!$, and how to increase the value of k when N decreases to $n!/k!$.
- How to increase the value of n, when N increases to $n!$.
- How to realize an efficient routing in the resulting network, including permutation routing, broadcasting, and multicasting.

References

1) S. Akers, D. Harel, and B. Krishnamurthy, "The Star Graph: An Attractive Alternative to the n-Cube," In *Proc. 1987 Int'l Conf. Parallel Process. (ICPP)* 1987, pages 393-400.
2) J. Aspnes and G. Shah, "Skip graphs," In *Proc. Annual ACM-SIAM Symp. on Discrete Algorithms (SODA)*, January 2003, pages 384-393.
3) H.-L. Chen and N.-F. Tzeng, "Enhanced Incomplete Hypercube," In *Proc. 1989 Int'l Conf. Parallel Process. (ICPP)*, vol. 1, 1989, pages 270-277.
4) T.-S. Chen and J.-P. Sheu, "Broadcasting on Incomplete Star Graph Interconnection Networks," *High Performance Computing on the Information Superhighway, 1997. HPC Asia '97*, April 1997, pages 67-72.
5) T.-S. Chen and N.-C. Wang, "Optimal Broadcasting on Incomplete Star Graph Interconnection Networks," *Journal of Systems Architecture*, 51(2): 143-150, February 2005.
6) Available at `http://www.gnutella.com/`.
7) D. Guo, J. Wu, H. Chen, and X. Luo, "Moore: An Extendable Peer-to-Peer Network Based on Incomplete Kautz Digraph with Constant Degree," In *Proc. IEEE Int'l Conf. Computer Communications (INFOCOM)*, May 2007.
8) T. Iwasaki and K. Kaneko, "A New Configuration of an Incomplete Star Graph and its Routing Algorithm," In *Proc. Int'l Conf. Parallel and Distributed Processing Techniques and Applications (PDPTA)*, July 2008, pages 116-122.
9) H. P. Katseff, "Incomplete Hypercubes," *IEEE Trans. Comput.*, 37(5): 604-608, May 1988.
10) C.P. Ravikumar, A. Kuchlous, and G. Manimaran, "Incomplete Star Graph: An Economical Fault-tolerant Interconnection Network," In *Proc. Int'l Conf. Parallel Process. (ICPP)*, August 1993, pages 83-90.
11) I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," *ACM SIGCOMM Computer Communication Review*, 31 (4): 149-160, 2001.
12) N.-F. Tzeng, "Structural Properties of Incomplete Hypercube Computers," In

Proc. 10th Int'l Conf. Distributed Computing Systems (ICDCS), May 1990, pages 262-269.

13) N.-F. Tzeng and H.-L. Chen, "Structural and Tree Embedding Aspects of Incomplete Hypercubes," *IEEE Trans. Comput.*, 43(12): 1434-1439, December 1994.

14) N.-F. Tzeng and G. Lin, "Efficient Determination of Maximum Incomplete Subcubes in Hypercubes with Faults," *IEEE Trans. Comput.*, 45(11): 1303-1308, November 1996.

15) N.-F. Tzeng and H. Kumar, "Traffic Analysis and Simulation Performance of Incomplete Hypercubes," *IEEE Trans. Parallel and Distributed Systems*, 7(7): 740-754, July 1996.