Fast and Dependable Communication in Hyper-rings

Tom Altman[1], Yoshihide Igarashi[2], and Kazuhiro Motegi[2]

[1] Department of Computer Science, University of Colorado at Denver, Denver, CO 80217, USA
taltman@carbon.cudenver.edu

[2] Department of Computer Science, Gunma University, Kiryu, Japan 376-8515
igarashi@comp.cs.gunma-u.ac.jp

Abstract. A graph $G = (V, E)$ is called a hyper-ring with N nodes (N-HR for short) if $V = \{0, ..., N-1\}$ and $E = \{\{u, v\} | v - u$ modulo N is a power of $2\}$. The following results are shown. We prove that the node-connectivity κ of an N-HR is equal to its degree, say δ, by presenting an algorithm for the explicit construction of δ node-disjoint paths connecting nodes s and t. The length of these paths is bounded by $\lceil \log D \rceil + 3$, where D is the positional distance between s and t. Finally, we show a node-to-node communication scheme for HRs that requires only $\lceil \log D \rceil + 3$ rounds, even in the presence of up to $\delta - 1$ node failures.

Keywords: Hyper-ring, connectivity, broadcasting, network, reliability

1 Introduction

Various topologies have been proposed for interconnecting processors in large scaled parallel and/or distributed systems. Hyper-rings (HRs) [2] and their variations have appeared in the literature under several names, including optimal broadcasting scheme [1] and binary jumping networks [8]. Detailed discussions on the structure, properties, and advantages of this family of networks and their fault tolerance were presented in [2,7,8].

Node-connectivity of a network is fundamental in the analysis of network reliability and/or security. Menger's Theorem relates the connectivity of a graph G, denoted by $\kappa(G)$, to the size of the smallest set (among the maximal sets) of pair-wise internally node-disjoint paths between any pair of nodes s and t [12]. Note that in any G, $\kappa \leq \delta$, where δ denotes the minimal nodal degree of G. Hence, in a network, multiple copies of a message may be sent through a number of disjoint paths and fault tolerance can be achieved in this manner [10].

In addition to increased reliability, disjoint paths provide an excellent mechanism for transmitting secret messages through a network. Using Rabin's Information Dispersal Algorithm [11], or partitioning a secret message into submessages, where every i-th packet is transmitted via the i-th mod κ path, may prevent (or at least make it more difficult) for adversaries to intercept and decipher it.

Han et al. [8], have extended the results obtained in [7] addressing the problem of broadcasting in faulty binary jumping networks, a directed version of

O.H. Ibarra and L. Zhang (Eds.): COCOON 2002, LNCS 2387, pp. 350–359, 2002.

HRs with roughly half the number of edges. However, their work was limited to the general broadcasting problem (one source sending the same message to all nodes in the network), and not to the more interesting problem of an explicit construction of the κ node-to-node disjoint paths, investigated here.

It is relatively easy to show that for any N-HR, $\delta - 2 \leq \kappa \leq \delta$, by applying the results of [4] to a *reduced* HR in which the maximal positional distance between any connected nodes is $2^{\lfloor \log N \rfloor - 1}$ and not the standard $2^{\lfloor \log N \rfloor}$. The reduced HR would be a circulant with the property of *convexity*, which is sufficient (but not a necessary) condition for $\kappa = \delta$ [4,5].

The major contribution of this paper is showing that HRs' connectivity is maximal by presenting an algorithm that generates the δ node-disjoint paths between any source node s and destination node t. Moreover, the length of all of these paths is bounded from above by $\lceil \log D \rceil + 3$, where $D \leq \lfloor N/2 \rfloor$ is the positional distance between s and t.

The rest of the paper is organized as follows. In Section 2, we present two fundamental methods of HR construction. There, we also introduce Semi Hyper-rings (SHRs) and enriched SHRs, discuss their connectivities, and node-disjoint path construction for SHRs. These will play a fundamental role in our node-to-node disjoint-path construction and communication schemes of Section 3.

2 Preliminaries

2.1 HR Construction

Let us begin with a formal definition of a hyper-ring.

Definition 1. A graph $G = (V, E)$ is called a *hyper-ring* with N nodes (N-HR for short) if $V = \{0, ..., N-1\}$ and $E = \{\{u, v\} | [v-u]_N$ is a power of 2 $\}$, where $[m]_r$ is m modulo r.

Definition 2. The *positional distance* between node s and node t of N-HR is defined to be $min\{[t-s]_N, [s-t]_N\}$.

The number of edges between nodes in an HR is roughly twice that of a hypercube (HC) with the same number of nodes, but the proposed organization possesses a number of advantages over the HC. In particular, for any N we can construct an N node HR, whereas a hypercube must contain exactly 2^k nodes for some k. An example of an 11-HR is shown in Figure 1. Note that what appears to be a connection of positional distance 3 is really a (counterclockwise) connection of positional distance 8. This class of edges, connecting nodes $2^{\lfloor \log N \rfloor}$ away from each other, will be addressed in more detail in the path construction procedures of this section.

The HCs have a natural and an elegant recursive structure that does not seem to extend to HRs in an obvious way. In particular, the number of edges in HRs does not grow monotonically with the number of nodes [2]. It turns out, however, that HRs do possess an interesting recursive construction that encompasses HCs

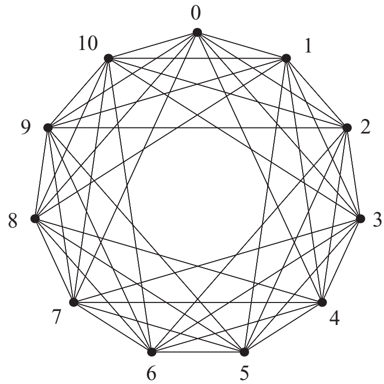

Fig. 1. An example of an 11-HR.

in a sense as a special case. examine First, let us observe a simple (nonrecursive) construction of N-HRs. The following procedure is straightforward and needs no further explanation.

> **for** $i := 0$ **to** $N-1$ **do**
>   **for** $k := 0$ **to** $\lfloor \log N \rfloor$ **do**
>   **if** there is no edge between i and $[i+2^k]_N$
>   **then** connect node i to nodes labeled $[i \pm 2^k]_N$

Observe that in HRs, the two edges usually connecting nodes that are either $2^{\lfloor \log N \rfloor}$ or $2^{\lfloor \log N \rfloor - 1}$ positional distance away from each other either do not exist as such (examine the $(2^i + 2^j)$-HR family), may merge into one (2^k-HR family), or crisscross (all of the remaining HRs). Again, see edges (0,8) and (0,$[-8]_{11} = 3$) in Figure 1.

Assume that an N-HR has already been constructed. Suppose we wish to construct a $2N$-HR from it. The following procedure [2], takes as input an N-HR_0 and returns a $2N$-HR.

> make a duplicate copy of N-HR_0, call it N-HR_1
> **for** $i := 0$ **to** $N-1$ **do**
>   relabel node i in N-HR_0 to $2i$
>   relabel node i in N-HR_1 to $2i+1$
> **for** $i := 0$ **to** $2N-1$ **do**
>   connect node i and node $[i+1]_{2N}$

Figure 2 shows the connections for one node of the HR constructed using the doubling construction procedure. It shows that if δ is the degree of an N-HR, $N > 1$, then the degree of $2N$-HR is $\delta + 2$. Previously, it was shown that the doubling of an HR increases its connectivity from κ to $\kappa + 2$ [3]. The essence of the proof is captured by the two paths containing nodes α and β in Figure 2, however, it will not be discussed here.

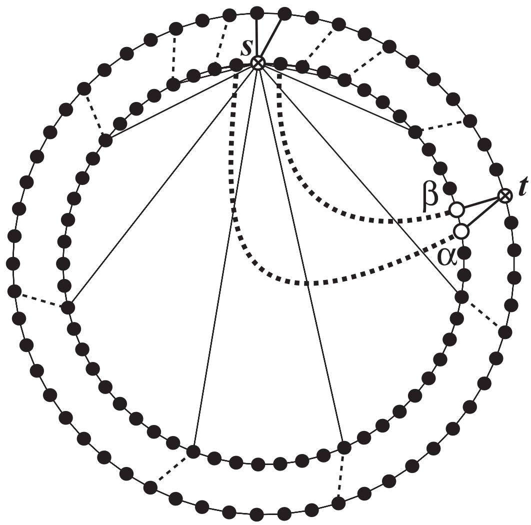

Fig. 2. An HR constructed using the doubling procedure and the two additional node-disjoint paths via α and β.

2.2 Semi Hyper-rings

Before we can proceed with the node-disjoint path construction algorithm for HRs, let us introduce a related structure and examine its connectivity and path algorithms.

Definition 3. A *Semi Hyper-ring* is a graph corresponding to any segment of P consecutive integers, (P-SHR for short) in which any two nodes (integers) are connected iff they are a power of 2 positional distance away from each other.

Lemma 1. The node-connectivity of a P-SHR is equal to $\lceil \log P \rceil$, which is also the degree of its first and last nodes.

Proof. It is obvious that the degree of the first and last nodes of any P-SHR is $\lceil \log P \rceil$. Observe that for all $r > 0$, the SHRs of size $2^r + 1$ through 2^{r+1} will have the same connectivity. This is due to the Expansion Lemma (see, e.g., [12]) which states that if G is a k-connected graph, and G' is obtained from G by adding a new vertex y with at least k neighbors in G, then G' is k-connected. This allows us to focus on SHRs of size $P = 2^q + 1$, e.g., the integers $0, 1, ..., 2^q$.

We show, by induction on q, that P-SHRs have connectivity of $\lceil \log P \rceil$. For $q = 1$, the 3-SHR is obviously 2-connected.

Inductive Hypothesis: Assume that any $(2^q + 1)$-SHR is $q + 1$-connected, i.e., between any two nodes s and t, there exist $q+1$ node-disjoint paths. Equivalently, if S_1 and S_2 are two disjoint sets of $q+1$ nodes each, all located within the same $(2^q + 1)$-SHR, then there exist $q + 1$ node-disjoint paths within it that pairwise connect the nodes of both sets.

Let us now examine a $(2^{q+1}+1)$-SHR. Observe that the degrees of all nodes in the $(2^{q+1}+1)$-SHR have increased by 1, e.g., the degree of node 0 is now $q+2$, etc. We must show that for any $0 \leq s < t \leq 2^{q+1}$ there are now $q+2$ node-disjoint paths between s and t. There are two cases to consider:

1. If $t-s \leq 2^q$, then by the Inductive Hypothesis there exist $q+1$ node-disjoint paths between s and t such that the internal nodes of these paths will be totally contained within a (2^q+1)-SHR whose one endpoint is either node s, or t, or an SHR whose endpoints are $t-2^q+1$ and $s+2^q-1$.
   The $(q+2\text{nd})$ path will have one of three possible forms, which will depend on the relative placement of s and t:
   $\{s, s-1, s-2, ..., 0, 2^{q+1}, 2^{q+1}-1, 2^{q+1}-2, ..., t+2^q, t\}$, if $s, t \leq 2^q$;
   $\{s, s-2^q, s-2^q-1, s-2^q-2, ..., 0, 2^{q+1}, 2^{q+1}-1, 2^{q+1}-2, ..., t\}$, if $s, t \geq 2^q$;
   $\{s, s+2^q, s+2^q+1, s+2^q+2, ..., 2^{q+1}, 0, 1, 2, ..., t-2^q, t\}$, if $s < 2^q < t$.
2. If $t-s > 2^q$, then by the Inductive Hypothesis and the Expansion Lemma there already exist $q+1$ node-disjoint paths between s and t. Furthermore, all of the internal nodes of these $q+1$ paths have indices greater than s and less than t. The $(q+2\text{nd})$ path will have the form: $\{s, s-1, s-2, ..., 0, 2^{q+1}, 2^{q+1}-1, 2^{q+1}-2, ..., t\}$. □

2.3 Path Construction for SHRs

For ease of presentation and clarity, let us visualize the problem of SHR/HR node-disjoint path construction as a simple (one player) blue/green pebble game, which has only three rules:

1. The only pebble movements (hops) allowed are via the existing edges.
2. A pebble may not be placed on a node previously visited by any other pebble.
3. If node i is occupied by a pebble, the player may *capture* it by hopping a different color pebble to i and removing both pebbles from the game.

The object of the game is to maximize the number of captured pebbles, given some initial pebble configuration. It should be clear that the traces of the pebble hops uniquely define a set of node-disjoint paths and that a capture indicates a completion of a path between the nodes where the pebbles were originally placed.

Let us now present procedure *CompressSHR* that given t as input generates the $\lceil \log t \rceil$ node-disjoint paths between 0 and t within the $(t+1)$-SHR. We will describe the steps of this procedure in terms of pebble movements.

procedure *CompressSHR(t)*
Initialize: place $\lceil \log t \rceil$ blue and $\lceil \log t \rceil$ green pebbles in locations $\{2^0, 2^1, ..., 2^{\lfloor \log t \rfloor}\}$ and $\{t-2^0, t-2^1, ..., t-2^{\lfloor \log t \rfloor}\}$, respectively.

0. Check for any green and blue pebbles placed on the same node, (including blue on t and green on 0) and if found, capture them.

while any pebbles are left **do**
1. Identify the green pebble g that is closest and to the right of the rightmost blue pebble b.
2. For all of the remaining green pebbles to the right of b use an appropriate *single-hop* to move them to the left of b.
   If any land on another blue pebble, consider it a capture.
3. Construct the appropriate sequence of hops for g to capture b.

It is clear that the green pebble movements represent the node t to node s paths and that they are node-disjoint (see Figure 3). Observe that at each loop iteration at least one capture is made, limiting the number of loops to $\lceil \log t \rceil$, making the maximal number of hops made by any pebble to be no more than $\lceil \log t \rceil$. Furthermore, if for some reason (e.g., an incorrect initial placement of one of the blue pebbles) step 1 fails, then let g be the green pebble closest to the rightmost blue pebble and continue with step 3, for that particular while-loop iteration. Later, we will see how that may become necessary during the HR path construction.

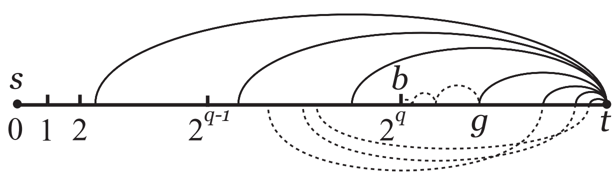

Fig. 3. Compression of an SHR and node-disjoint path construction.

Definition 4. A P-SHR is said to be *enriched* if it is extended by allowing additional connections between all pairs of nodes that are of k positional distance away from each other, for a fixed k, $3 \leq k < P$, which is not a power of 2.

Hence, the degree of the first and last nodes in any enriched P-SHR is equal to $\lceil \log P \rceil + 1$. Observe that procedure *CompressSHR*(t) can be naturally extended to include enriched SHRs by initially placing additional blue and green pebbles in locations k and $t - k$, respectively.

Observation 1 *Procedure CompressSHR will capture all of the pebbles in any (enriched) P-SHR.*

Let $s < t$ be two nodes in an N-HR that are $D \leq \lfloor N/2 \rfloor$ positions away from each other. We may construct a (possibly enriched) D-SHR$_{(s,t)}$ that spans s and t by removing from the N-HR all of the nodes (and their edges) outside the line segment (s,t), (i.e., outside $\{s, s+1, ..., t-1, t\}$).

Let us focus on the family of N-HRs that possess the crisscrossing edges connecting nodes of $2^{\lfloor \log N \rfloor}$ positional distance.

Observation 2 *If $D \geq N - 2^{\lfloor \log N \rfloor}$ then the degree of nodes s and t as well as the number of node-disjoint paths between s and t in the D-SHR$_{(s,t)}$ is equal to $\lceil \log D \rceil + 1$, otherwise it is $\lceil \log D \rceil$.*

The next result will allow us to obtain the disjoint-path length bounds for not only SHRs, but eventually, HRs.

Lemma 2. *For any (enriched) D-SHR, the length of the node-disjoint paths constructed by procedure CompressSHR is bounded by $\lceil \log D \rceil + 2$.*

3 Path Construction for HRs

We now present our algorithm for the construction of node-disjoint paths in HRs. First, given N, s, and t we will make the initial pebble placement by putting the blue (and green) pebbles on nodes which are exactly $\pm 2^i, i = 0, ..., \lfloor \log N \rfloor$, positions away from s (and t), respectively (making any immediate capture(s), if any blue/green pair of pebbles has been found on the same node). Observe that a placement of a blue pebble on t and a green one on s indicates that s and t are directly connected by an edge and calls for an immediate capture of the two pebbles.

Let D be the minimal (clockwise, counterclockwise) positional distance between s and t and $d = \lceil \log D \rceil$. Denote by M_s and M_t the nodes $[s + \lfloor N/2 \rfloor]_N$ and $[t - \lfloor N/2 \rfloor]_N$, respectively.

Next, as shown in Figure 4, the following six SHRs will be identified. Note that, in certain extreme cases, (e.g., $2K$-HRs and $D = K$), SHR$_B$, SHR$_{B'}$, SHR$_C$, and SHR$_{C'}$ may contain no nodes, and s and t may serve as end nodes for both SHR$_A$ and SHR$_E$.

1. SHR$_A$ is the segment (s, t) of size $2 \leq D \leq \lfloor N/2 \rfloor$.
2. SHR$_B$ is bounded by $s - 1$ and the closer of two nodes $(s - 2^d,\ M_t + 1)$.
   SHR$_{B'}$ is bounded by $t + 1$ and the closer of two nodes $(t + 2^d,\ M_s - 1)$.
3. SHR$_C$ is bounded by closer of two nodes $(s - 2^d - 1,\ M_t + 1)$, and $M_t + 1$.
   SHR$_{C'}$ is bounded by closer of two nodes $(t + 2^d + 1,\ M_s - 1)$, and $M_s - 1$.
4. SHR$_E$ is bounded by M_t and M_s.

We can easily capture all of the pebbles in SHR$_A$ by *CompressSHR*.

Next, we will capture the pebbles from the nodes of SHR$_B$ and SHR$_{B'}$. We single-hop the blue pebbles from SHR$_B$ to SHR$_{B'}$ by taking their 2^{d+1} (clockwise) connections. While it is clear that all of the blue pebbles would wind up in SHR$_{B'}$, there are two potential problems.

1. One blue pebble may already be in SHR$_{B'}$ via the 2^d-connection. To address this problem, one of the blue pebbles should be left behind in SHR$_B$, the question of which one is answered below.
2. One blue pebble may already be in SHR$_{B'}$ via the $2^{\lfloor \log N \rfloor}$-connection. If so, a second blue pebble would have to be left behind in SHR$_B$.

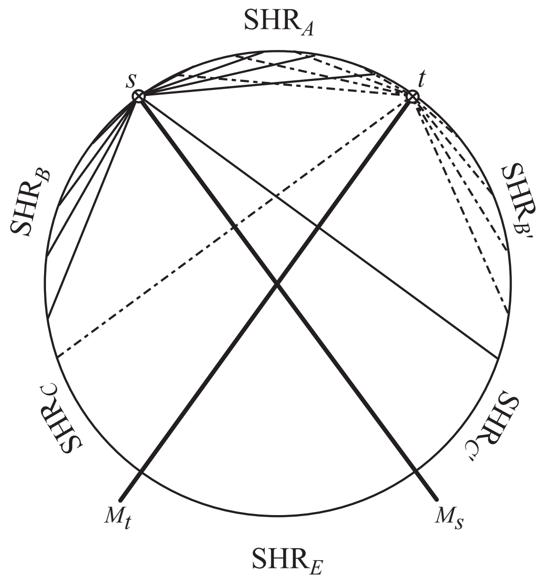

Fig. 4. Partitioning of an N-HR.

A quick check would determine if any of the SHR$_B$ blue pebbles could land on the above-mentioned (blue) problem pebbles in SHR$_{B'}$ (and those would be the ones that would stay behind in SHR$_B$. If no such pebble(s) exist in SHR$_B$, then we pick them randomly. In any case, at most $d+1$ blue (and the same number of green) pebbles would be present in SHR$_{B'}$ after this 2^{d+1} cross-over. Observe that the one/two blue pebble(s) left behind will now be used within SHR$_B$ to join the one/two green pebbles that (by symmetry) would have been in SHR$_B$ from the beginning. At this time *CompressSHR* would be used on SHR$_{B'}$.

We now turn our attention to SHR$_C$ and SHR$_{C'}$. Observe that if $D \neq 2^d$, the pebbles in SHR$_C$ are in locations (blue) $2^{d+1}, 2^{d+2}, \ldots$ and (green) $2^{d+1} - D, 2^{d+2} - D, \ldots$, positions away from s. The blue/green pebble pairs in SHR$_C$ from locations $(2^{d+j}, 2^{d+j} - D)$ away from s will be appropriately moved and captured. The positional distance between the blue and green pebbles within these pairs is bounded by D. The possible presence of the pebbles (via the $2^{\lfloor \log N \rfloor}$-connection) is not a significant problem here, since we are joining at most two pairs of pebbles within at least a 2D-sized subsegments of SHR$_C$. The blue/green pebble pairs in SHR$_{C'}$ would be captured in a symmetrical fashion.

If $D = 2^d$, then we would proceed as above, except there could be two *lonely* pebbles left: a blue pebble, b, located 2^{d+1} positions away from s in SHR$_C$ and a green pebble, g, located 2^{d+1} positions away from t in SHR$_{C'}$. These will be paired-up by allowing b to cross-over into SHR$_{C'}$ via a 2^{d+2} connection. Of course a direct hop from b's original position is not possible since it would collide with another blue pebble's location (the 2^{d+1} edge of s). Therefore, b could take a mini-hop of distance 2^{d-1} toward s and then take the cross-over edge 2^{d+2}. The (potential) problem in the case when the mini-hop would land b on a location of blue pebble corresponding to the $2^{\lfloor \log N \rfloor}$ connection from s can be resolved easily by making the mini-hop of distance 2^{d-2} instead. Once in $\mathrm{SHR}_{C'}$, b and g would be less than D distance away from each other and would be captured. Alternatively, b and g may be moved into SHR_E and captured there.

Finally, the pebbles from SHR_E would be captured using *CompressSHR*. Observe that as D approaches $\lfloor N/2 \rfloor$, SHR_E will have more and more pebbles. In fact, for $2K$-HRs and $D = K$, SHR_A and SHR_E are both K-SHRs with identical pebble distribution.

The above steps, summarized below, form procedure *Hop-a-log* that will construct the node-disjoint (s, t)-paths in a given HR.

procedure *Hop-a-log(N,s,t)*
  Make the initial pebble placement and an immediate capture check
  Partition the HR into the six SHRs
  Compress SHR_A
  Identify which and cross-over the blue pebbles from SHR_B to $\mathrm{SHR}_{B'}$
  Compress $\mathrm{SHR}_{B'}$
  Capture the one/two remaining pebble pairs in SHR_B
  Capture the $(2^{d+j}, 2^{d+j} - D)$ pairs in SHR_C and $\mathrm{SHR}_{C'}$
  If they exist, capture the two *lonely* pebbles using the 2^{d+2} hop
  Compress SHR_E.

Lemma 3. *For any HR the* Hop-a-log *procedure will generate $\kappa = \delta$ node-disjoint paths between any source node s and destination node t. Moreover, the length of each such path is bounded by $\lceil \log D \rceil + 3$.*

Proof. Follows from proof of Lemma 1, Observation 2, and Lemma 2, and the fact that the sizes of all of the SHRs on which *CompressSHR* was used, were bounded by D. □

Theorem 1. *A node-to-node communication scheme using procedure* Hop-a-log *will send a message from s to t within $\lceil \log D \rceil + 3$ rounds, even in the presence of $\kappa - 1$ node failures.*

4 Concluding Remarks

Besides facilitating a fast and reliable node-to-node communication, HRs provide an excellent means for network broadcasting. Assuming a multi-port broadcasting model in which a node can send a message to all of its immediate neighbors in one round, broadcasting in HRs may be carried out in only $\lceil \log N \rceil$ rounds to send information from any source to all destination nodes, if no nodes have failed. The bound on the path lengths in Lemma 3, however, guarantees that an HR broadcasting scheme using this procedure will send a message from any source to all destinations within $\lceil \log N \rceil + 2$ rounds, even in the presence of $\kappa - 1$ node failures.

Hyper-rings appear to be the first log-sparse connection architecture that allows for dependable communication in time of $\lceil \log D \rceil + 3$, where D is the positional distance between the source and destination nodes. This *speed of locality* is especially important for fast and dependable none-to-node communication protocols and distributed computing. Examples of such applications would include distributed pattern recognition algorithms, battlefield communication management, monitoring systems, etc. HRs are certainly worthy of further studies.

References

1. N. Alon, A. Barak, and U. Mauber, On disseminating information reliably without broadcasting, *7th Int. Conference on Distributed Computer Systems* (1987) 74-81.
2. T. Altman, Y. Igarashi, and K. Obokata, Hyper-ring connection machines, *Parallel Computing* 21 (1995) 1327-1338.
3. T. Altman, Reliable communication schemes for hyper-rings, *28th Int. Southeastern Conference on Combinatorics, Graph Theory, and Computing* (1997).
4. F. Boesch and A. Felzer, A general class of invulnerable graphs, *Networks* 2 (1972) 261-283.
5. F. Boesch and R. Tindell, Circulants and their connectivities, *J. of Graph Theory* 8 (1984) 487-499.
6. E. van Doorn, Connectivity of circulant digraphs, *J. of Graph Theory* 10 (1986) 9-14.
7. Y. Han, R. Finkel, An optimal scheme for disseminating information, *The 22nd Int. Conference on Parallel Processing* (1988) 198-203.
8. Y. Han, Y. Igarashi, K. Kanai, and K. Miura, Broadcasting in faulty binary jumping networks, *J. of Parallel and Distributed Computing* 23 (1994) 462-467.
9. W. Knödel, New gossips and telephones, *Discrete Mathematics* 13 (1975) 95.
10. A. Pelc, Fault-tolerant broadcasting and gossiping in communication networks, *Networks* 28 (1996) 143-156.
11. M. O. Rabin, Efficient dispersal of information for security, load balancing, and fault tolerance, *J. of ACM* 36 (1989) 335-348.
12. D. West, *Introduction to Graph Theory,* Prentice Hall (1996).