PAPER

Fault-Tolerant Routing Algorithms for Hypercube Interconnection Networks

Keiichi KANEKO† *and* **Hideo ITO**††, ***Regular Members***

SUMMARY Many researchers have used hypercube interconnection networks for their good properties to construct many parallel processing systems. However, as the number of processors increases, the probability of occurrences of faulty nodes also increases. Hence, for hypercube interconnection networks which have faulty nodes, several efficient dynamic routing algorithms have been proposed which allow each node to hold status information of its neighbor nodes. In this paper, we propose an improved version of the algorithm proposed by Chiu and Wu by introducing the notion of full reachability. A fully reachable node is a node that can reach all nonfaulty nodes which have Hamming distance l from the node via paths of length l. In addition, we further improve the algorithm by classifying the possibilities of detours with respect to each Hamming distance between current and target nodes. We propose an initialization procedure which makes use of an equivalent condition to perform this classification efficiently. Moreover, we conduct a simulation to measure the improvement ratio and to compare our algorithms with others. The simulation results show that the algorithms are effective when they are applied to low-dimensional hypercube interconnection networks.
key words: *hypercube interconnection networks, fault-tolerant routing, full reachability, hamming distance, communication*

1. Introduction

Recently, interest in parallel processing is spreading rapidly and many parallel processing systems have appeared. Current researchers have a tendency to target the parallel processing based on a connection of extreme numbers of processors, that is, massively parallel processing. However, as the number of processors increases, the probability of occurrences of faulty nodes also increases. Hence, it is necessary to construct communication paths which detour faulty processors.

We adopt a hypercube interconnection network [5], [8], [9] as the target parallel processing system. Many researchers have used hypercube networks for their good properties, such as a symmetric and regular structure and a relatively small diameter [1], [6], [10]. We focus on a dynamic routing scheme for message communication between processors in a hypercube network which has multiple faulty nodes in order to suppress degradation of system performance after generation of faulty nodes.

Manuscript received December 7, 1999.
Manuscript revised July 13, 2000.
†Faculty of Technology, Tokyo University of Agriculture and Technology, Koganei-shi, 184–8588 Japan.
††Faculty of Engineering, Chiba University, Chiba-shi, 263–8522 Japan.

In a parallel processing system which has faulty processors, it is very important to select one of the shortest paths to the target node to establish communication between processors. If every processor in the system were to identify the status of all other processors, an optimal routing would be possible. However, because of restrictions of space and time complexities to solve the shortest path problem, it is very difficult to adopt this approach. For hypercube interconnection networks, several efficient dynamic routing algorithms have been proposed which allow each node to hold status information of neighbor nodes [3], [4], [7], [11]. One of the algorithms we propose is based on Chiu and Wu's [3], with the addition of the notion of full reachability to improve performance. In addition, we further introduce the notion of unsafe nodes with respect to distance to improve the algorithm. Finally, we conducted a simulation to measure the improvement ratio and to compare our algorithms with an algorithm by Chiu and Chen [4] which uses the equivalent notion of the proposed full reachability.

The rest of this paper is constructed as follows. Section 2 defines several preliminary definitions and describes the routing algorithm proposed by Chiu and Wu [4]. In Sect. 3 and 4, we give definitions of full reachability and node classifications. In addition, the proposed algorithms and initialization procedures are presented. In Sect. 5, a simulation is performed and the algorithms are evaluated. Section 6 describes conclusions.

2. Routing Algorithms

2.1 Hypercube Interconnection Networks

First of all, we give the definition of a hypercube.

Definition 1: (d-dimensional hypercube)
A d-dimensional hypercube interconnection network consists of 2^d nodes whose addresses are represented by d-bit binary numbers. A node n has d adjacent nodes whose addresses are obtained by reverting, one by one, each bit of its address.

Let $H(n_1, n_2)$ be the Hamming distance between two nodes n_1 and n_2 in a hypercube; then the length of the shortest path between two nodes n_1 and n_2 is equal to $H(n_1, n_2)$ if there is no faulty node.

Consider a message delivery from a source node s to a target node t. First, let $N(s)$ represent the set of neighbor nodes of the node s.

$$N(s) = \{n \mid H(s,n) = 1\}$$

Next, let $D(s,t)$ represent the subset of neighbor nodes of s which are closer to the target node t than the node s.

$$D(s,t) = \{n \mid H(n,t) = H(s,t) - 1, n \in N(s)\}$$

Then, the equation $|D(s,t)| = H(s,t)$ holds and we can send the message with the address of target node t to any node c in $D(s,t)$. Now, let us consider the node c as a new source one, and repeat the process above until the message reaches the target node.

2.2 Routing Problem

In a hypercube interconnection network which has faulty nodes, it is necessary for message delivery to find a path which goes from the source node to the target and touches no faulty nodes. For this purpose, each node can store information about neighbor nodes and make use of that combined with the address of the target node to select a neighbor node dynamically to send the message. In addition, even if any shortest path cannot be found by the information, a detour must be detected. An algorithm which performs these operations is called a routing algorithm. In this situation, a good algorithm finds as many shortest paths as possible while holding as simple information as possible.

Definition 2: (reachability and communicability) If there exists a path from the source node s to the target node t which includes no faulty nodes, t is said to be reachable from s. If a routing algorithm R finds the path, t is said to be communicable from s by R.

2.3 Algorithm by Chiu and Wu

In this section, we describe the algorithm proposed by Chiu and Wu [3] which is referred as `route` in this paper. The algorithm, first, divides the nonfaulty nodes in a hypercube network into safe and unsafe nodes. It then classifies the unsafe nodes into ordinary and strongly unsafe nodes. Here are their main definitions and theorems followed by the algorithm.

The first definition gives a classification of nonfaulty nodes. Note that it is defined recursively.

Definition 3: (safe and unsafe nodes) A nonfaulty node n is unsafe if it is adjacent to two or more faulty nodes or it is adjacent to more than two faulty or unsafe nodes. A nonfaulty node is safe if it is not unsafe.

Definition 4: (strongly and ordinary unsafe nodes) An unsafe node n is strongly unsafe if every neighbor node of n is either unsafe or faulty. An unsafe node n is ordinary unsafe if it is not a strongly unsafe node.

Figure 1 shows a classification example of nodes based on the definitions above. Moreover, we define full unsafeness as a property of hypercube interconnection networks.

Definition 5: (fully unsafe networks) A hypercube interconnection network is fully unsafe if all the nonfaulty nodes in the network are unsafe nodes.

In the rest of this paper, let S, $\bar{U}$ and $\tilde{U}$ represent the set of safe, ordinary unsafe and strongly unsafe nodes, respectively. Figure 2 shows the algorithm by Chiu and Wu.

Then the following theorems hold according to the property of hypercube interconnection networks [3].

Theorem 1: If either the source node s or the target t is safe, the algorithm `route` can communicate by one of the shortest paths of length $H(s,t)$.

Theorem 2: If the source node s is ordinary unsafe and the target t is unsafe, the algorithm `route` can communicate by a path whose length is at most $H(s,t)+2$.

Theorem 3: In a hypercube interconnection network

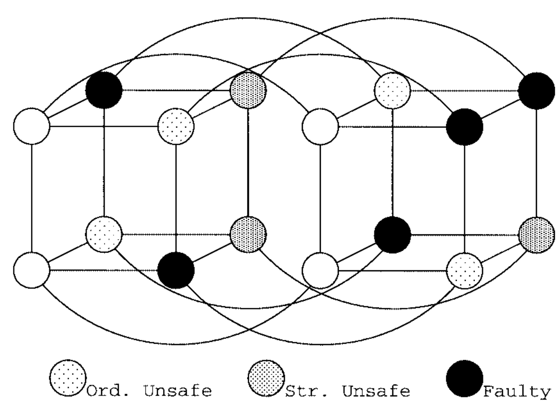


Fig. 1 Node classification by Chiu and Wu.

```
procedure route(c, t)
begin
 l := H(c,t); N := N(c); D := D(c,t);
 if l = 0 then deliver the message to c and exit
 else if ∃n ∈ D ∩ S then nxt := n
 (* for future replacement *)
 else if ∃n ∈ D ∩ Ū then nxt := n
 else if ∃n ∈ D ∩ Ũ and (c ∈ Ũ or l ≤ 2) then
       nxt := n
 else if ∃n ∈ (N − D) ∩ S then nxt := n
 else if ∃n ∈ (N − D) ∩ Ū then nxt := n
 else error('unable to deliver');
 route(nxt, t)
end
```

Fig. 2 Routing algorithm `route` by Chiu and Wu.

which is not fully unsafe, every strongly unsafe node is adjacent to an ordinary unsafe node. Hence, if the source node s is strongly unsafe and the target t is an unsafe node in a hypercube which is not fully unsafe, the algorithm `route` can communicate by a path whose length is at most $H(s,t)+4$.

Each nonfaulty node in a hypercube interconnection network exchanges information with its neighbor nodes to classify itself as a safe, an ordinary unsafe, or a strongly unsafe node. Using this classification, if a d-dimensional hypercube interconnection network is not fully unsafe, an effective routing can be implemented based on theorems 1, 2, and 3. Even if the hypercube network is fully unsafe, the algorithm `route` is applicable. However, the target is not always communicatable by the algorithm even if it is reachable from the source. In this case, it is necessary to switch to other worse algorithms [2].

3. An Algorithm Based on Full Reachability

3.1 Full Reachability and Safe Nodes with Respect to Distance

This section describes the algorithm `FR` which we propose. It makes use of the fact that the Hamming distance to the target is available to select the neighbor node to send a message. For this purpose, a property of reachability is defined: a node is reachable to every nonfaulty node which is apart from it by some fixed Hamming distance via a path of the same length as the Hamming distance.

Definition 6: (full reachability with respect to distance)
A nonfaulty node n is fully reachable with respect to (Hamming) distance h, if every nonfaulty nodes which is apart from the node n by Hamming distance h is reachable from the node n via a path of length h.

Let R_h represent the set of nodes which are fully reachable with respect to distance h. When a node n which is apart from the target node by Hamming distance $h+1$ receives a message and tries to select a node to send it, if it could be known whether every one of its neighbor nodes belongs to R_h or not, unnecessary detours could be avoided. However, the membership of a nonfaulty node to R_h depends on the distribution of all those faulty nodes which are apart from it by a Hamming distance of h or less. It means that each nonfaulty node must collect the information about all faulty nodes to decide if it itself belongs to R_h or not. Therefore, it is difficult to identify R_h and find an appropriate route. To address this problem, we introduce an approximation of R_h.

Definition 7: (safe nodes with respect to distance)
Every nonfaulty node is a safe node with respect to (Hamming) distance 1. A nonfaulty node is a safe node with respect to distance h if it is adjacent to more than or equal to $d-h+1$ safe nodes with respect to distance $h-1$.

In the rest of this paper, let S_h represent the set of safe nodes with respect to distance h. For example, consider a 4-dimensional hypercube network in which each node is addressed as shown in Fig. 3 and the set of the faulty nodes is $\{1,4,12,13,14\}$. Then following equations hold:

$$\begin{aligned}
R_1 &= S_1 = \{0,2,3,5,6,7,8,9,10,11,15\},\\
R_2 &= \{2,3,6,7,8,10,11,15\},\\
S_2 &= \{2,3,7,8,10,11\},\\
R_3 &= \{0,2,3,6,7,9,10,11,15\},\\
S_3 &= \{0,2,3,6,9,10,11,15\}.
\end{aligned}$$

The node 15 is fully reachable with respect to distance 2 because each node (i.e. $3,5,6,9$, and 10) which is nonfaulty and apart from 15 by Hamming distance 2 is reachable from 15 via a path of length 2. However, it is not a safe node with respect to distance 2. Similarly, the node 7 is fully reachable with respect to distance 3 though it is not safe with respect to distance 3.

3.2 Algorithm FR

About S_h and R_h, the following theorem holds.

Theorem 4: For any distance h, $S_h \subset R_h$.
(Proof) The theorem is proved by induction on h. Let $n \in S_1$, then n is not a faulty node. Therefore, for any nonfaulty node n' for which $H(n,n')=1$ holds, that is, for every nonfaulty neighbor node n', n is reachable to it immediately using the directly connected edge. Hence, $S_1 \subset R_1$. Now, assume that $S_h \subset R_h$ holds for every $h<k$. Here, let $n \in S_k$ then from Definition 7, the node n must be adjacent to at least $d-k+1$ nodes which are safe with respect to distance $k-1$ (see Fig. 4). This means that the number of neighbor nodes which are not safe with respect to distance $k-1$ is at most $k-1$. For

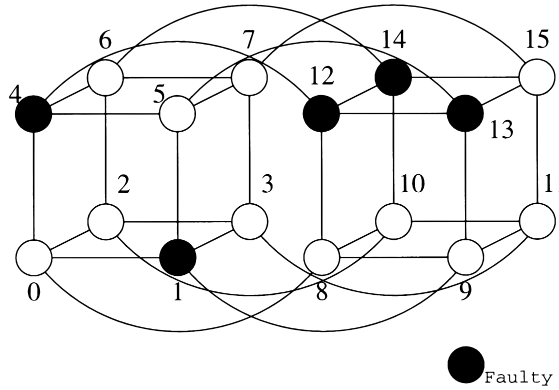


Fig. 3 An example of 4-cube with faulty nodes.

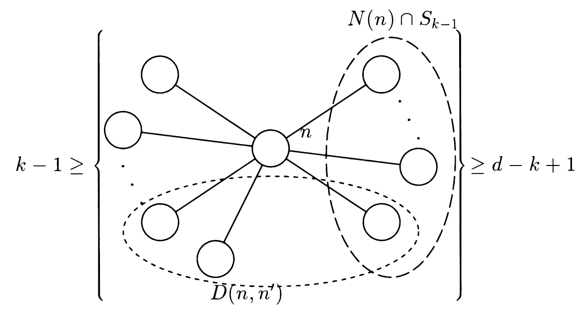


Fig. 4 Relationship between node $n(\in S_k)$ and its neighbor nodes.

any nonfaulty node n' for which $H(n,n') = k$ holds, consider the set $D(n,n')$ whose elements are neighbor nodes of n and nearer to n' than n. The number of nodes included in $D(n,n')$ is k ($|D(n,n')| = k$). Therefore, at least one node of k nodes in $D(n,n')$ is safe with respect to distance $k-1$. From the hypothesis of induction, $S_{k-1} \subset R_{k-1}$. Then if we send a message to the node, it is possible to reach the target node n' through a path of length k. Hence, for any nonfaulty node n' for which $H(n,n') = k$ holds, n is reachable to n' using a path of length k and $n \in R_k$. From the above $S_k \subset R_k$, and for any distance h, $S_h \subset R_h$ holds. □

Generally, it is difficult to obtain $N(n) \cap R_h$ for any node n, while S_h is easily detected because it can be determined by exchanging information just between neighbor nodes. By Theorem 4, $S_h \subset R_h$ holds. Hence we can use S_h as a set of safe nodes to send messages. If we calculate $S_2, \cdots, S_k$ in a preprocess, a new routing algorithm FR is obtained by changing the comment line (*···*) of Fig. 2 with
else if $l \le k+1$ and $\exists n \in D \cap S_{l-1}$ then nxt := n
(see Fig. 5).

As shown in the following Theorem 5, the algorithm FR results in more nodes which can be used to route safely than does the algorithm route.

Theorem 5: For any distance h, $S \subset S_h$.
(Proof) The theorem is proved by induction on h. $S \subset S_1$ is trivial. If $n \in S$ then the node n is adjacent to at most one faulty node and $n \in S_2$ from Definition 7. Hence $S \subset S_2$. Now, assume $S \subset S_h$ holds for any $h < k$. Here, let $n \in S$; then n is adjacent to at most two non-safe nodes from Definition 3. Therefore, n is adjacent to at least $d-2$ safe nodes ($|N(n) \cap S| \ge d-2$). From the hypothesis of induction, $S \subset S_{k-1}$. Hence, $N(n) \cap S \subset N(n) \cap S_{k-1}$ holds and this means that the node n is adjacent to at least $d-2$ nodes which are safe with respect to distance $k-1$ ($|N(n) \cap S_{k-1}| \ge d-2$). Now, $d-2 \ge d-k+1$ because $k \ge 3$. Then, the node n is adjacent to at least $d-k+1$ nodes which are safe with respect to distance $k-1$. This means $n \in S_k$ from Definition 7, hence $S \subset S_k$. Therefore, for any distance h, $S \subset S_h$ holds. □

```
procedure FR(c, t)
begin
 l := H(c,t); N := N(c); D := D(c,t);
 if l = 0 then deliver the message to c and exit
 else if ∃n ∈ D ∩ S then nxt := n
 else if l ≤ k + 1 and ∃n ∈ D ∩ S_{l−1} then
       nxt := n
 else if ∃n ∈ D ∩ Ū then nxt := n
 else if ∃n ∈ D ∩ Ũ and (c ∈ Ũ or l ≤ 2) then
       nxt := n
 else if ∃n ∈ (N − D) ∩ S then nxt := n
 else if ∃n ∈ (N − D) ∩ Ū then nxt:= n
 else error('unable to deliver');
 FR(nxt, t)
end
```

Fig. 5 Routing algorithm FR based on full reachability.

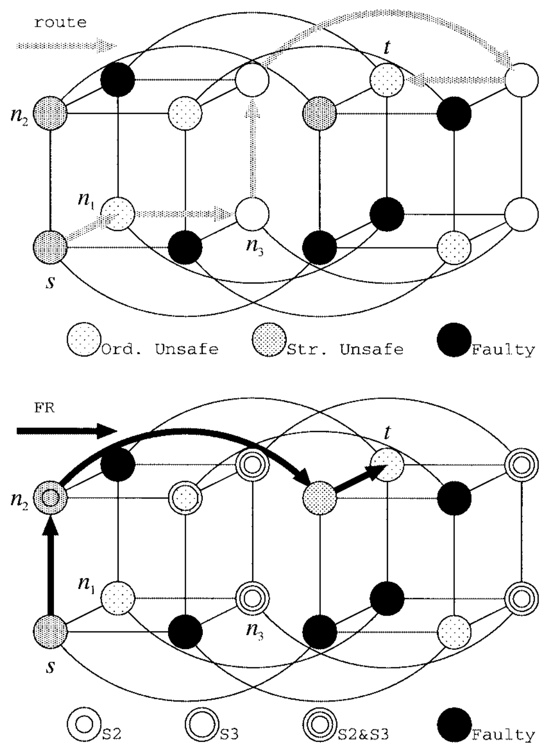


Fig. 6 A routing example by route and FR.

In the example shown in Fig. 6, since the source node s is strongly unsafe, the algorithm route sends the message to a single ordinary unsafe node n_1 in $D(s,t)$, that is, the subset of neighbor nodes of s which are closer to the target node t than the node s. However, the nodes in $D(n_1,t)$ are all faulty and it must detour to a safe node n_3. In contrast, the algorithm FR finds that the node n_2 in $D(s,t)$ which is judged strongly unsafe in route is safe with respect to distance 2. Therefore, it can construct a path shown by arrows by using information about full reachability based on Hamming distance.

```
procedure init(c, k)
begin
 σ_{c,1} := SAFE;
 Detect N(c) ∩ S_1;
 for h := 2 to k do
  begin
   send σ_{c,h-1} to N(c) ∩ S_1;
   for every n ∈ N(c) ∩ S_1 do receive σ_{n,h-1} from n;
   T_{h-1} := {n|n ∈ N(c) ∩ S_1, σ_{n,h-1} = SAFE}
   if |T_{h-1}| ≥ d - h + 1 then σ_{c,h} := SAFE
   else σ_{c,h} := UNSAFE
  end
end
```

Fig. 7 Initialization procedure `init` for routing algorithm `FR`.

3.3 Initialization for `FR`

Next, we show an algorithm to accumulate in each nonfaulty node information about its neighbors in a hypercube interconnection network which has faulty nodes. We presume that every node has buffers, one for each link between a neighbor node and itself and that constant time is required for sending and receiving a message. In addition, we make the assumption that the detection of faulty neighbor nodes can be performed in constant time.

Figure 7 shows the initialization procedure `init` for node c in `FR`. A variable $\sigma_{n,h}$ holds classification information about a node n with respect to distance h. A variable T_{h-1} represents a subset of neighbor nodes of c which are safe with respect to $h-1$, and the value $\sigma_{c,h}$ is determined according to its cardinality.

This procedure must be executed in addition to the procedure of initialization for the algorithm `route` whose time complexity is $O(d^2)$ [3]. But time complexity for the procedure `init` is $O(kd)$, it does not make the whole time complexity worse.

4. An Algorithm Based on Classification of Unsafe Nodes

4.1 Classification of Unsafe Nodes with Respect to Distance

If we can assign k as equal to $d-1$ in the algorithm `FR`, it is not necessary to use the classification information of safe nodes by Chiu and Wu for routing selection. Similarly, it is possible to route based only on node classification with respect to Hamming distance by classifying unsafe nodes with respect to distance.

Definition 8: (unsafe nodes with respect to distance) A nonfaulty node n is unsafe with respect to (Hamming) distance h if the node n is not safe with respect to distance h.

That is, if the number of neighbor nodes of a nonfaulty node n which are faulty or unsafe with respect to distance $h-1$ is greater than or equal to h, then the node n is unsafe with respect to distance h.

In addition, to detect a subset of unsafe nodes with respect to distance which gives guaranteed detours, we introduce a definition below.

Definition 9: (strongly and ordinary unsafe nodes with respect to distance)
For an unsafe node n with respect to distance h, consider an arbitrary division of the neighbor nodes of n into two disjoint subsets N_1 and N_2 which consist of h and $d-h$ nodes, respectively. The node n is ordinary unsafe with respect to distance h if, for all such divisions, the subset N_1 includes a safe node with respect to distance $h-1$ or a safe node with respect to distance $h+1$ belongs to the subset N_2. If a node n which is unsafe with respect to distance h is not an ordinary unsafe node with respect to distance h, it is a strongly unsafe node with respect to distance h.

A node n which is ordinary unsafe with respect to distance h has the following property for a target node t apart from n by Hamming distance $h (= H(n,t))$: the node n has a safe node with respect to distance $h-1$ in $D(n,t)$, the subset of neighbor nodes which are closer to the target node than the node n, or it has a safe node with respect to distance $h+1$ in $N(n) - D(n,t)$ to detour.

4.2 Initialization for `FR2`

It is difficult to detect ordinary unsafe nodes with respect to distance by using Definition 9 directly. Hence we make use of the following theorem.

Theorem 6: For an unsafe node n with respect to distance h, the node n is ordinary unsafe with respect to distance h if and only if at least one of the following conditions holds:

- There exists a node n' among the neighbor nodes of n which is safe with respect to distance $h+1$ and $h-1$.
- The number of safe nodes with respect to distance $h+1$ among the neighbor nodes of n is greater than or equal to $h+1$.

(Proof) First comes the proof of sufficiency. Let a node n be unsafe with respect to distance h. Now, let us consider the case that there exists a node n' among the neighbor nodes of n which is safe with respect to distance $h+1$ and $h-1$. In Definition 9, since n' is a neighbor node of n, if n' belongs to the subset N_1 then there exists a node in N_1 which is safe with respect to distance $h-1$; otherwise n' belongs to N_2 and there exists a node in N_2 which is safe with respect to distance $h+1$. Next, consider the case that the number of safe nodes with respect to distance $h+1$ among the neighbor nodes of n is greater than or equal to $h+1$. Then, since $|N_1| = h$, there exists a safe node in N_2 with

```
procedure init2($c$)
begin
 $\sigma_{c,1}$ := SAFE;
 Detect $N(c) \cap S_1$;
 for $h$ := 2 to $d$ do
  begin
   send $\sigma_{c,h-1}$ to $N(c) \cap S_1$;
   for every $n \in N(c) \cap S_1$ do receive $\sigma_{n,h-1}$ from $n$;
   $T_{h-1}$ := $\{n | n \in N(c) \cap S_1, \sigma_{n,h-1} = \mathtt{SAFE}\}$;
   if $|T_{h-1}| \geq d-h+1$ then $\sigma_{c,h}$ := SAFE
   else $\sigma_{c,h}$ := S_UNSAFE
  end;
 send $\sigma_{c,d}$ to $N(c) \cap S_1$;
 for every $n \in N(c) \cap S_1$ do receive $\sigma_{n,d}$ from $n$;
 $T_d$ := $\{n | n \in N(c) \cap S_1, \sigma_{n,d} = \mathtt{SAFE}\}$;
 for $h$ := 2 to $d-1$ do
  if $\sigma_{c,h}$ = S_UNSAFE then
   if $\exists n \in T_{h-1} \cap T_{h+1}$ or $|T_{h+1}| \geq h+1$ then
    $\sigma_{c,h}$ := O_UNSAFE;
end
```

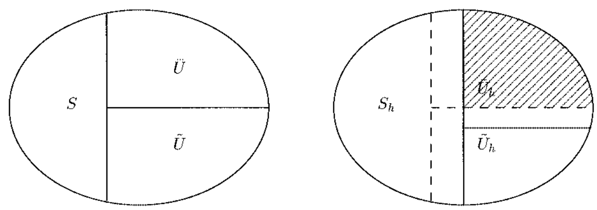
Fig. 8 Initialization procedure init2 for routing algorithm FR2.

respect to distance $h+1$. Consequently, the node n is proved to be ordinary unsafe with respect to distance h in either case. Secondly, we prove the necessity. For an ordinary unsafe node n with respect to distance h, we assume that there is no node in neighbor nodes of n which is safe with respect to distance $h+1$ and $h-1$, and the number of safe nodes with respect to distance $h+1$ in neighbor nodes of n is less than or equal to h. Then, we can divide $N(n)$ into N_1 and N_2 where all safe nodes with respect to distance $h+1$ are included in N_1 and N_2 includes all the safe nodes with respect to distance $h-1$ of neighbor nodes of n. Hence, n is not ordinary unsafe with respect to distance h which is a contradiction. The theorem is proved by the above discussion. □

By using Theorem 6, the algorithm shown in Fig. 8 can classify the neighbor nodes of each nonfaulty node c; where as similar to the initialization procedure for the algorithm FR, we assume that each node has buffers, each of which is situated at the link between its neighbor node and itself; message sending and receiving are performed in constant time; and faulty neighbor nodes are detectable in constant time. A variable $\sigma_{n,h}$ holds the classification information of node n with respect to distance h and a variable T_h represents the subset of neighbor nodes of c which are safe with respect to distance h. The value of $\sigma_{c,h}$ is determined according to the variable T_h.

4.3 Algorithm FR2

In the rest of this paper, let $\bar{U}_h$ and $\tilde{U}_h$ represent the set of ordinary unsafe nodes with respect to distance h and that of strongly unsafe nodes with respect to distance h, respectively. Concerning the classification above, Theorems 7 and 8 hold.



(a) route (b) FR2

Fig. 9 Node classification with respect to distance.

```
procedure FR2($c$, $t$)
begin
 $h$ := H($c$,$t$); $N$ := N($c$); $D$ := D($c$,$t$);
 if $h = 0$ then deliver the message to $c$ and exit
 else if $\exists n \in D \cap S_{h-1}$ then $nxt$ := $n$
 else if $\exists n \in D \cap \bar{U}_{h-1}$ then $nxt$ := $n$
 else if $\exists n \in D \cap \tilde{U}_{h-1}$ and ($c \in \tilde{U}_h$ or $h \leq 2$) then
      $nxt$ := $n$
 else if $\exists n \in (N-D) \cap S_{h+1}$ then $nxt$:=$n$
 else if $\exists n \in (N-D) \cap \bar{U}_{h+1}$ then $nxt$:=$n$
 else error('unable to deliver');
 FR2($nxt$, $t$)
end
```

Fig. 10 Algorithm FR2 based on classification of unsafe nodes.

Theorem 7: For any Hamming distance h, $S \cup \bar{U} \subset S_h \cup \bar{U}_h$.
(Proof) Because S is a subset of S_h, it is sufficient to show that the set $\bar{U} - S_h$(the hatched part in Fig. 9) is a subset of $\bar{U}_h$. Any node n which belongs to $\bar{U} - S_h$ is included $\bar{U}$. Hence n has a safe node (according to the classification by Chiu and Wu) n' in its neighbor nodes. Therefore, n' is safe with respect to distance $h-1$ and $h+1$. From Theorem 6, the node n is ordinary unsafe with respect to distance h. Consequently, $\bar{U} - S_h \subset \bar{U}_h$. □

Theorem 8: For any distance h, $\tilde{U}_h \subset \tilde{U}$.
(Proof) It is obvious from Theorem 7. □

From Theorem 8, if a hypercube interconnection network is not fully unsafe (by Definition 5), a strongly unsafe node with respect to distance h has an ordinary unsafe neighbor node. From this fact, we can construct a new algorithm FR2 based on the classification of unsafe nodes. See Fig. 10.

5. Evaluation

To verify the power of our algorithms FR and FR2, we repeat the following procedure for all combinations of addresses of faulty nodes and a target one. For the algorithm FR, we adopted a best parameter value of $k = d-1$.

1. In a d-dimensional hypercube, set f faulty nodes.
2. Classify all nodes into faulty, safe, ordinary unsafe, and strongly unsafe nodes. Moreover, calculate S_h, $\bar{U}_h$ and $\tilde{U}_h$, $(1 \leq h \leq d)$.

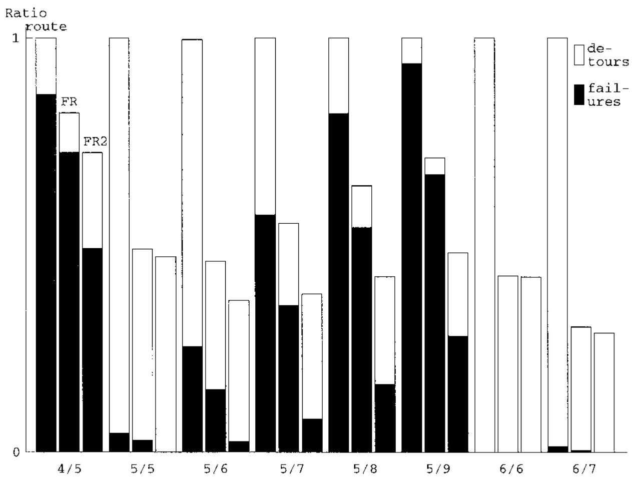


Fig. 11 Improvement ratio by algorithm FR2.

3. Taking advantage of symmetry, fix the node 0 as the source, and select a nonfaulty node which is not the source and is reachable from the source node as the target.
4. Call the procedures for `route`, `FR`, and `FR2`, then count the number of unnecessary detours and failures of deliveries for each one.

Figure 11 shows the results. The horizontal axis d/f represents pairs of the dimension d of the hypercube interconnection network and the number of faulty nodes f. Specifically, for any given d/f pair, it represents the ratio of the sum of the number of detours and the number of failures in our algorithms `FR` and `FR2`, to the similar sum obtained for `route` algorithm.

In every case, `FR2` shows the best results and `FR` is better than `route`. Focusing on the number of failures, `FR2` reduces it further than `FR` does. We believe this is due to the new scheme based on the classification of unsafe nodes with respect to distance.

Moreover, we compare our algorithms `FR` and `FR2` with an algorithm `RC` by Chiu and Chen [4]. Their algorithm uses the notion of routing capability which is equivalent to our full reachability. Assume that the current and target nodes are c and t, respectively. Then the algorithm searches for a node to proceed or detour to in $D \cap S_{h-1}$, $D \cap S_{h+1}$, $(N - D) \cap S_{h+1}$, $D \cap S_{h+3}$, $(N - D) \cap S_{h+3}$, $\cdots$ in this order where $h = H(c, t)$, $N = N(c)$ and $D = D(c, t)$. Though the directed version of classification is applicable to all algorithms including ours, it is ignored for simplicity. Simulation is performed by following the procedure mentioned above with one exception, namely, that the addresses of faulty nodes and a target node are randomly generated one million times. Figure 12 and Fig. 13 show the average percentage of the shortest path routing and the average reachability of the algorithms in a 6-dimensional hypercube, respectively. The reachability is the ratio of messages which managed to reach the target nodes.

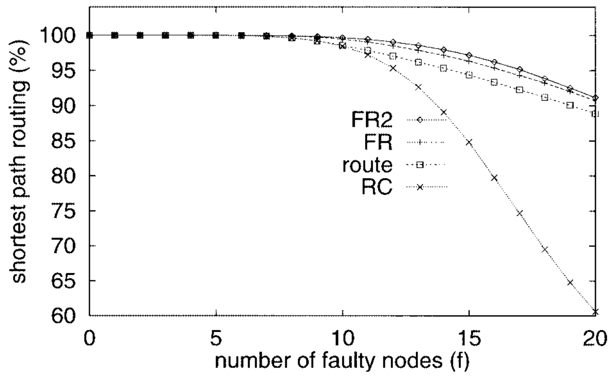


Fig. 12 Shortest path routing percentage in 6-cube.

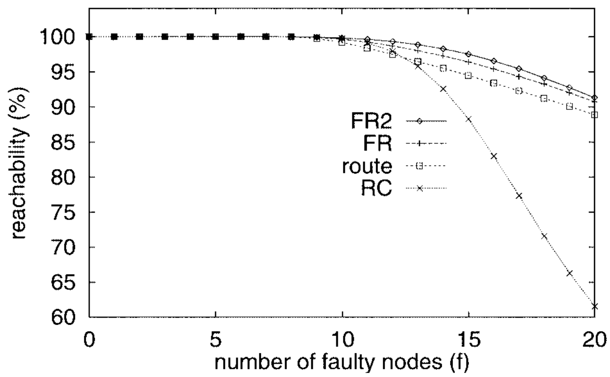


Fig. 13 Reachability percentage in 6-cube.

In either case, results show that the algorithms `FR1` and `FR2` are superior to `RC`. The reason of this, we believe, is that `RC` uses only the routing capabilities while others use the notion of unsafeness for a better speculative routing which ensures reachability.

6. Conclusions

We first proposed a routing algorithm `FR` which is based on full reachability and is an extension of the algorithm by Chiu and Wu [3]. We further proposed another routing algorithm `FR2` which does not make use of classification information used in Chiu and Wu [3] by classifying unsafe nodes with respect to the Hamming distance.

Evaluation of the algorithms shows that they can detect communication paths which do not include any faulty nodes and which were not found by conventional algorithms `route` and `RC`. In addition, it is demonstrated by computer simulation that `FR` and `FR2` are effective for low-dimensional hypercubes and give good results.

In future, it is necessary to execute simulations to assess the applicability of `FR2` for high-dimensional hypercube interconnection networks.

References

[1] J. Bruck, R. Cypher, and D. Soroker, "Embedding cube-connected cycles graphs into faulty hypercubes," IEEE Trans. Comp., vol.43, no.10, pp.1210–1220, Oct. 1994.
[2] M.-S. Chen and K.G. Shin, "Depth-first search approach for fault-tolerant routing in hypercube multicomputers," IEEE Trans. Parallel and Distributed Systems, vol.1, no.2, pp.152–159, April 1990.
[3] G.-M. Chiu and S.-P. Wu, "A fault-tolerant routing strategy in hypercube multicomputers," IEEE Trans. Comp., vol.45, no.2, pp.143–155, Feb. 1996.
[4] G.-M. Chiu and K.-S. Chen, "Use of routing capability for fault-tolerant routing in hypercube multicomputers," IEEE Trans. Comp., vol.46, no.8, pp.953–958, Aug. 1997.
[5] W.J. Dally, "Performance analysis of k-ary n-cube interconnection networks," IEEE Trans. Comp., vol.39, no.6, pp.775–785, June 1990.
[6] J. Kim and C.R. Das, "Hypercube communication delay with wormhole routing," IEEE Trans. Comp., vol.43, no.7, pp.806–814, July 1994.
[7] T.C. Lee and J.P. Hayes, "A fault-tolerant communication scheme for hypercube computers," IEEE Trans. Comp., vol.41, no.10, pp.1242–1256, Oct. 1992.
[8] C.L. Seitz, "The cosmic cube," CACM, vol.28, no.7, pp.22–33, July 1985.
[9] Y. Saad and M.H. Schultz, "Topological property of hypercubes," IEEE Trans. Comp., vol.37, no.7, pp.867–872, July 1988.
[10] N.-F. Tzeng and H.-L. Chen, "Structural and tree embedding aspects of incomplete hypercubes," IEEE Trans. Comp., vol.43, no.12, pp.1434–1439, Dec. 1994.
[11] J. Wu, "Adaptive fault-tolerant routing in cube-based multicomputers using safety vectors," IEEE Trans. Parallel and Distributed Systems, vol.9, no.4, pp.322–334, April 1998.

Keiichi Kaneko is an Associate Professor at Tokyo University of Agriculture and Technology. His main research areas are functional programming, parallel and distributed computation, partial evaluation and fault-tolerant systems. He received the B.E., M.E. and Ph.D. degrees from the University of Tokyo in 1985, 1987 and 1994, respectively. He is also a member of ACM, IPSJ, and JSSST.

Hideo Ito is a Professor at Chiba University. His research interests include easily-testable design, test generation, VLSI architecture and fault-tolerant VLSI design. He received the B.E. degree from Chiba University in 1969, and the D.E. degree from Tokyo Institute of Technology in 1984. He is also a member of IPSJ and IEEE Computer Society.