Fault-tolerant Routing based on Directed Safety Levels in a Hyper-Star Graph

Yo Nishiyama Yuki Hirai Keiichi Kaneko
Department of Computer and Information Sciences
Tokyo University of Agriculture and Technology
Tokyo, Japan
Email: k1kaneko@cc.tuat.ac.jp

***Abstract*—A hyper-star graph *HS*(2*n*, *n*) is promising as a generic topology for interconnection networks of parallel processing systems because it has merits of a hypercube and a star graph. This paper proposes an algorithm that constructs a fault-free path between a pair of non-faulty nodes in *HS*(2*n*, *n*) with faulty nodes. In addition, we conduct a computer experiment to show its effectiveness.**

Keywords-interconnection network; hypercube; star graph; faulty nodes

I. Introduction

With rapid research progress on parallel processing systems, many new topologies [1], [2], [3], [4], [5] for interconnection networks have been proposed instead of simple topologies such as rings, meshes, tori, hypercubes [6], and so on. A hyper-star graph $HS(m, n)$ is one such topology, and it is promising as a generic topology, which provides interconnection networks for parallel processing systems, because it has advantages of a hypercube and a star graph simultaneously [7]. In a parallel processing system, it is necessary to construct an algorithm which assumes existence of faulty elements. Therefore, in this paper, we focus on a regular hyper-star graph $HS(2n, n)$ that has faulty nodes and propose an adaptive fault-tolerant routing algorithm between any pair of non-faulty nodes.

If each non-faulty node in a graph collects status of all other nodes as global information, an optimal fault-tolerant routing is possible. However, this method requires the same order of memory space and communication time as the number of the graph. Hence, it is impractical. On the other hand, if each non-faulty node executes fault-tolerant routing by collecting status of its neighbor nodes only as local information, high reachability cannot be achieved. Therefore, in a hypercube, there are several researches that try to attain high reachabilities by collecting a part of global information as restricted global information [8], [9], [10], [11], [12], [13]. In addition, Nishiyama et al. introduced an index called safety level for a hyper-star graph and proposed a fault-tolerant routing algorithm where restricted global information is collected to achieve high reachability [14]. In this paper, we propose a new fault-tolerant routing algorithm by introducing a new notion of directed safety level to attain higher reachability. The directed safety level is obtained by improving the safety level by Nishiyama et al.

The rest of this paper is structured as follows. First, in Section 2, a definition of a hyper-star graph and other necessary definitions are introduced, and the algorithm by Nishiyama et al. is explained. Next, in Section 3, we explain our algorithm in detail. Then, we verify effectiveness of our algorithm by a computer experiment in Section 4. Finally, we mention a conclusion and a future work in Section 5.

II. Preliminaries

In this section, at first, we give definitions of a regular hyper-star graph and safety levels.

Definition 1: A regular hyper-star graph $HS(2n,n)$ is an undirected graph, which has ${}_{2n}C_n$ nodes. Each node $\boldsymbol{a}$ consists of $2n$ bits $(a_1, a_2, \ldots, a_{2n})$, among which n bits are 1 while remaining n bits are 0 ($\boldsymbol{a} \in \{0, 1\}^{2n}$, $\sum_{i=1}^{2n} a_i = n$). For two nodes $\boldsymbol{a} = (a_1, a_2, \ldots, a_{2n})$ and $\boldsymbol{b} = (b_1, b_2, \ldots, b_{2n})$, there exists an edge $(\boldsymbol{a}, \boldsymbol{b})$ between them if and only if there exists $k (\in \{2, 3, \ldots, 2n\})$ such that $b_1 = \underline{a}_1$, $b_k = \underline{a}_k = a_1$, and $b_i = a_i$ $(2 \leq i \neq k \leq 2n)$. Then we use the notation $\boldsymbol{b} = \boldsymbol{a}^{(k)}$.

Figure 1 shows an example of $HS(6,3)$. In addition, Table II shows comparison among a hyper-star graph $HS(2n, n)$, a hypercube Q_n, a star graph S_n, a perfect hierarchical hypercube HHC_{2^n+n}, and a hierarchical cube network HCN_n.

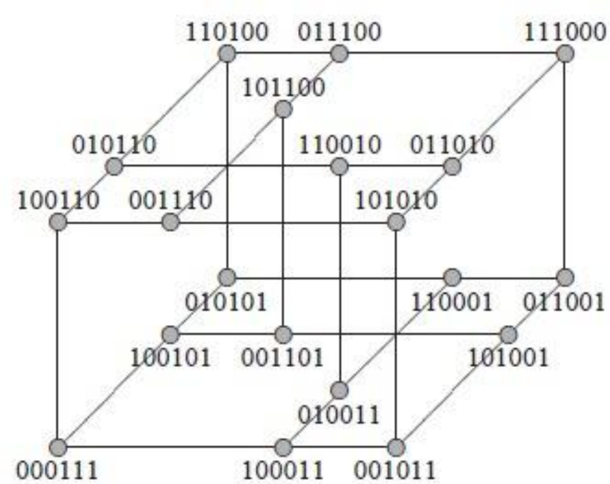

Figure 1. A hyper-star graph $HS(6, 3)$

978-0-7695-4959-0/13 $25.00 © 2013 IEEE
DOI 10.1109/ACSAT.2012.76

Table I

COMPARISON OF A HYPER-STAR GRAPH WITH OTHER TOPOLOGIES.

	#nodes	degree	connect.	diameter
$HS(2n, n)$	${}_{2n}C_n$	n	n	$2n-1$
Q_n	2^n	n	n	n
S_n	$n!$	$n-1$	$n-1$	$\lfloor 3(n-1)/2 \rfloor$
HHC_{2^n+n}	2^{2^n+n}	$n+1$	$n+1$	2^{n+1}
HCN_n	2^{2n}	$n+1$	$n+1$	$\lfloor 4(n+1)/3 \rfloor$

For two nodes $\boldsymbol{a} = (a_1, a_2, \ldots, a_{2n})$ and $\boldsymbol{b} = (b_1, b_2, \ldots, b_{2n})$ in $HS(2n,n)$, their distance $d(\boldsymbol{a}, \boldsymbol{b})$ is defined as $\sum_{i=2}^{2n} a_i \oplus b_i$. Moreover, among neighbor nodes of $\boldsymbol{a}$, those which are closer to $\boldsymbol{b}$ than $\boldsymbol{a}$ are denoted by $Pre(\boldsymbol{a}, \boldsymbol{b}) = \{\boldsymbol{n} \mid \boldsymbol{n} \in N(\boldsymbol{a}), d(\boldsymbol{n}, \boldsymbol{b}) = d(\boldsymbol{a}, \boldsymbol{b}) - 1\}$ while those which are further from $\boldsymbol{b}$ than $\boldsymbol{a}$ are denoted by $Spr(\boldsymbol{a}, \boldsymbol{b}) = \{\boldsymbol{n} \mid \boldsymbol{n} \in N(\boldsymbol{a}), d(\boldsymbol{n}, \boldsymbol{b}) = d(\boldsymbol{a}, \boldsymbol{b}) + 1\}$.

Then, the following Lemma 1 holds.

Lemma 1: For two node $\boldsymbol{a} = (a_1, a_2, \ldots, a_{2n})$ and $\boldsymbol{b} = (b_1, b_2, \ldots, b_{2n})$ with $d = d(\boldsymbol{a}, \boldsymbol{b})$ in $HS(2n, n)$, $a_1 = b_1$ if d is even, and $a_1 = \underline{b}_1$ if d is odd.

For a node $\boldsymbol{a} = (a_1, a_2, \ldots, a_{2n})$ in an $HS(2n, n)$, we define a set of indices $I(\boldsymbol{a}) = \{i \mid 2 \le i \le 2^n, a_i = \underline{a}_1\}$. Note that $|I(\boldsymbol{a})| = n$.

For a node $\boldsymbol{a}$ in $HS(2n, n)$, Nishiyama et al. introduced the safety level $S_d(\boldsymbol{a})$ [14] as an indicator to ensure that a fault-free path of length d can be established for an arbitrary non-faulty node with distance d from the node $\boldsymbol{a}$.

Definition 2: For a node $\boldsymbol{a}$ in $HS(2n,n)$ with a set of faulty nodes F, a safety level $S_d(\boldsymbol{a})$ ($0 \le d \le 2n-1$) with respect to distance d is defined as follows. That is, if $a \in F$, $S_d(\boldsymbol{a}) = 0$ for any d. If $\boldsymbol{a} \notin F$, $S_d(\boldsymbol{a}) = 1$ in case that either of the following conditions holds, and $S_d(\boldsymbol{a}) = 0$ otherwise.

1) $d \le 1$.
2) $2 \le d \le 2n-1$. For any $J(\subset I(\boldsymbol{a}))$ such that $|J| = \lceil d/2 \rceil$, there exists $j(\in J)$ such that $S_{d-1}(\boldsymbol{a}^{(j)}) = 1$.

If we calculate the safety levels for each distance following the definition, it takes much time. Hence, we introduce the next lemma by which we can calculate the safety levels easier.

Lemma 2: For a node $\boldsymbol{a}$ in $HS(2n,n)$ with a set of faulty nodes F, If $\boldsymbol{a} \notin F$, $2 \le d \le 2n-1$, the condition 2) in the definition of the safety level $S_d(\boldsymbol{a})$ of a node $\boldsymbol{a}$ with respect to distance d is equivalent to $|\{ j \mid S_{d-1}(\boldsymbol{a}^{(j)}) = 1, j \in I(\boldsymbol{a})\}| \ge n - \lceil d/2 \rceil + 1$.

(Proof) If $|\{ j \mid S_{d-1}(\boldsymbol{a}^{(j)}) = 1, j \in I(\boldsymbol{a})\}| \ge n - \lceil d/2 \rceil + 1$ holds, then for any $J(\subset I(\boldsymbol{a}))$ such that $|J| = \lceil d/2 \rceil$, there exists $j(\in J)$ such that $S_{d-1}(\boldsymbol{a}^{(j)}) = 1$. Therefore, necessity is proved. On the other hand, if $|\{ j \mid S_{d-1}(\boldsymbol{a}^{(j)}) = 1, j \in I(\boldsymbol{a})\}| < n - \lceil d/2 \rceil + 1$, then for any $j(\in J)$, there exists J such that $S_{d-1}(\boldsymbol{a}^{(j)}) = 0$, and the condition 2) does not hold. Hence, sufficiency is also proved.

From Lemma 2, for a non-faulty node $\boldsymbol{a}$ and a distance d from the destination node, the safety level is obtained by calculating $\sum_{j \in I(\boldsymbol{a})} S_{d-1}(\boldsymbol{a}^{(j)})$ and comparing it with $n - \lceil d/2 \rceil + 1$ to judge satisfaction of the condition 2). Figure 2 shows the algorithm that calculates the safety levels $S_d(\boldsymbol{a})$ ($d = 1,2,\ldots,2n-1$) for a node $\boldsymbol{a}$ in $HS(2n, n)$ based on this method as a procedure `SL`.

```
procedure SL(a, F)
begin
  for d := 0 to 2n−1 do
    if a ∈ F then S_d(a) := 0
    else if d <= 1 then S_d(a) := 1
    else if Σ_{j∈I(a)} S_{d−1}(a^(j)) >= n − ⌈d/2⌉ + 1 then
         S_d(a) := 1
    else S_d(a) := 0
end
```

Figure 2. Calculation algorithm for safety levels.

Figure 3 shows the algorithm for fault-tolerant routing based on the safety levels in $HS(2n, n)$ with a set of faulty nodes F as a procedure FT. In order to send a message from a non-faulty node $\boldsymbol{s}$ to a non-faulty node $\boldsymbol{d}$, we can call the procedure by `FT` ($\boldsymbol{s}$, $\boldsymbol{d}$, F).

```
procedure FT (c, d, F)
begin
  d := d(c, d);
  if d = 0 then deliver the message to c
  else begin
    n_p* := argmax_{n∈Pre(c,d)}{S_{d−1}(n)};
    n_s* := argmax_{n∈Spr(c,d)}{S_{d+1}(n)};
    if S_{d−1}(n_p*) = 1 then FT(n_p*, d, F)
    else if S_{d+1}(n_s*) = 1 then FT(n_s*, d, F)
    else if ∃n_p* ∈ Pre(c, d)\F then
      FT(n_p*, d, F)
    else if ∃n_s* ∈ Spr(c, d)\F then
      FT(n_s*, d, F)
    else error('delivery failed')
  end
end
```

Figure 3. Fault-tolerant routing algorithm.

With respect to the time complexity of calculation of the safety levels, it can be completed in $O(n^2)$ time in each node from the theorem below.

Theorem 1: In each node in $HS(2n, n)$, the time complexity to calculate the safety levels with respect to all distances is $O(n^2)$.

(Proof) From Lemma 2, in order to calculate a safety level $S_d(\boldsymbol{a})$ in a node $\boldsymbol{a}$, it is necessary to collect $S_{d-1}(\boldsymbol{n})$ from each node $\boldsymbol{n}$ which belongs to the neighbor node set $N(\boldsymbol{a})$ and calculate their sum. Hence, $O(n)$ time complexity is required.

Therefore, to calculate the safety levels with respect to all d ($2 \le d \le 2n - 1$), it takes $O(n^2)$ time complexity in total.

III. Proposed Algorithm

In the method by Nishiyama et al., if the condition does not hold critically, that is, if $\sum_{j \in I(\boldsymbol{a})} S_{d-1}(\boldsymbol{a}^{(j)}) = n - \lceil d/2 \rceil$, then for a neighbor node $\boldsymbol{a}^{(k)}$ of $\boldsymbol{a}$ such that $S_{d-1}(\boldsymbol{a}^{(k)}) = 0$, for any $J(\subset I(\boldsymbol{a}) \setminus \{k\})$ such that $|J| = \lceil d/2 \rceil$, there exists j ($\in J$) such that $S_{d-1}(\boldsymbol{a}^{(j)}) = 1$. Therefore, for the neighbor node $\boldsymbol{a}^{(k)}$, the node $\boldsymbol{a}$ seems as if it is a node with safety level 1 ($S_d(\boldsymbol{a}) = 1$) though actually $S_d(\boldsymbol{a}) = 0$ holds. See Figure 4. In this section, we give a detailed explanation of the method that includes this idea.

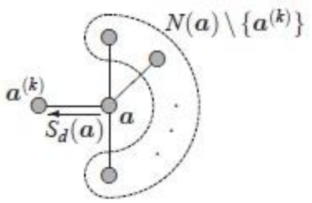

Figure 4. Safety level and neighbor nodes.

For a node $\boldsymbol{a}$ in $HS(2n, n)$, we introduce the directed safety level $\vec{S}_d^{(k)}(\boldsymbol{a})$ as a new indicator to ensure that a fault-free path of length d can be established for an arbitrary non-faulty node with distance d from the node $\boldsymbol{a}$ if the message is sent from the neighbor node $\boldsymbol{a}^{(k)}$ of $\boldsymbol{a}$.

Definition 3: For a node $\boldsymbol{a}$ in $HS(2n, n)$ with a set of faulty nodes F, a directed safety level $\vec{S}_d^{(k)}(\boldsymbol{a})$ ($0 \le d \le 2n - 1$) with respect to distance d and direction k is defined as follows. That is, if $\boldsymbol{a} \in F$, $\vec{S}_d^{(k)}(\boldsymbol{a}) = 0$ for any d and k. If $\boldsymbol{a} \notin F$, $\vec{S}_d^{(k)}(\boldsymbol{a}) = 1$ in case that either of the following conditions holds, and $\vec{S}_d^{(k)}(\boldsymbol{a}) = 0$ otherwise.

1) $d \le 1$.
2) $2 \le d \le 2n - 2$. For any $J(\subset I(\boldsymbol{a}) \setminus \{k\})$ such that $|J| = \lceil d/2 \rceil$, there exists $j(\in J)$ such that $\vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) = 1$.
3) $d = 2n - 1$. There exists $j(\in I(\boldsymbol{a}))$ such that $\vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) = 1$.

If we calculate the directed safety levels for each distance following the definition, it takes much time. Hence, we introduce a simple calculation method, which is based on the next lemma.

Lemma 3: For a node a in $HS(2n, n)$ with a set of faulty nodes F, if $\boldsymbol{a} \notin F$, $2 \le d \le 2n - 2$, the condition 2) in the definition of the directed safety level $\vec{S}_d^{(k)}(\boldsymbol{a})$ of a node $\boldsymbol{a}$ with respect to distance d and direction k is equivalent to $|\{ j \mid \vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) = 1, j \in I(\boldsymbol{a}) \setminus \{k\}\}| \ge n - \lceil d/2 \rceil$.

(Proof) If $|\{ j | \vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) = 1, j \in I(\boldsymbol{a}) \setminus \{k\}\}| \ge n - \lceil d/2 \rceil$, then for any $J(\subset I(\boldsymbol{a}))$ such that $|J| = \lceil d/2 \rceil$, there exists $j(\in J)$ such that $\vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) = 1$. Therefore, necessity is proved. On the other hand, if $|\{ j \mid \vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) = 1, j \in I(\boldsymbol{a}) \setminus \{k\}\}| < n - \lceil d/2 \rceil$, for any j ($\in J$) there exists J such that $\vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) = 0$, and the condition 2) does not hold. Hence, sufficiency is also proved.

From Lemma 3, for a non-faulty node $\boldsymbol{a}$, a distance d from the destination node, and a direction k, the directed safety level is obtained by calculating $\sum_{j \in I(\boldsymbol{a})} \vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) - \vec{S}_{d-1}^{(k)}(\boldsymbol{a}^{(k)})$ and comparing it with $n - \lceil d/2 \rceil$ to judge satisfaction of the condition 2). Figure 5 shows the algorithm that calculates the directed safety levels $\vec{S}_d^{(k)}(\boldsymbol{a}^{(k)})$ ($d = 1,2,\ldots,2n-1$) for a node $\boldsymbol{a}$ in $HS(2n, n)$ based on this method as a procedure DSL.

```
procedure DSL(a, F)
begin
  for δ := 0 to 2n − 1 do
    if a ∈ F then
      for each k ∈ I(a) do S⃗_d^(k)(a) := 0
    else if d <= 1 then
      for each k ∈ I(a) do S⃗_d^(k)(a) := 1
    else if d = 2n − 1 then
      if Σ_{j∈I(a)} S⃗_{d−1}^(j)(a^(j)) >= 1 then
        for each k ∈ I(a) do S⃗_d^(k)(a) := 1
      else
        for each k ∈ I(a) do S⃗_d^(k)(a) := 0
    else begin
      s := Σ_{j∈I(a)} S⃗_{d−1}^(j)(a^(j));
      for each k ∈ I(a) do
        if s − S⃗_{d−1}^(k)(a^(k)) >= n − ⌈d/2⌉ then
          S⃗_d^(k)(a) := 1
        else S⃗_d^(k)(a) := 0
    end
end
```

Figure 5. Calculation algorithm for safety levels.

Figure 6 shows the algorithm for fault-tolerant routing based on the directed safety levels in $HS(2n, n)$ with a set of faulty nodes F as a procedure DFT. For two adjacent nodes $\boldsymbol{a}$ and $\boldsymbol{b}$, $\delta(\boldsymbol{a}, \boldsymbol{b})$ represents the value that makes the equation $\boldsymbol{a}^{(\delta(\boldsymbol{a},\, \boldsymbol{b}))} = \boldsymbol{b}$ hold. In order to send a message from a non-faulty node $\boldsymbol{s}$ to a non-faulty node $\boldsymbol{d}$, we can call the procedure by DFT ($\boldsymbol{s}$, $\boldsymbol{d}$, F).

```
procedure DFT (c, d, F)
begin
  d := d(c, d);
  if d = 0 then deliver the message to c
  else begin
    n*_p := argmax_{n∈Pre(c,d)} { S^{δ(c,n)}_{d−1}(n) };
    n*_s := argmax_{n∈Spr(c,d)} { S^{δ(c,n)}_{d+1}(n) };
    if S^{δ(c,n*_p)}_{d−1}(n*_p) = 1 then DFT(n*_p, d, F)
    else if S^{δ(c,n*_s)}_{d+1}(n*_s) = 1 then DFT(n*_s, d, F)
    else if ∃n*_p ∈ Pre(c, d)\F then
      DFT(n*_p, d, F)
    else if ∃n*_s ∈ Spr(c, d)\F then
      DFT(n*_s, d, F)
    else error('delivery failed')
  end
end
```

Figure 6. Proposed fault-tolerant routing algorithm

With respect to the time complexity of calculation of the directed safety levels, it can be completed in $O(n^2)$ time in each node from the theorem below. The complexity is equal to that of the safety levels proposed by Nishiyama et al.

Theorem 2: In each node in $HS(2n, n)$, the time complexity to calculate the directed safety levels with respect to all distances and all directions is $O(n^2)$.
(Proof) From Lemma 3, in order to calculate a directed safety level $\vec{S}_d^{(k)}(\boldsymbol{a})$ in a node $\boldsymbol{a}$, it is sufficient to calculate $\sum_{j \in I(\boldsymbol{a})} \vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)}) - \vec{S}_{d-1}^{(k)}(\boldsymbol{a}^{(k)})$ and compare it with $n - \lceil d/2 \rceil$. $\sum_{j \in I(\boldsymbol{a})} \vec{S}_{d-1}^{(j)}(\boldsymbol{a}^{(j)})$ is calculated only once for each d. Hence, it takes $O(n^2)$ time complexity in total. A comparison can be done in $O(1)$ time complexity. Hence, it can be done for all d and all k in $O(n^2)$ time complexity. Therefore, it takes $O(n^2)$ time complexity to calculate the directed safety levels with respect to all d and all k.

IV. Evaluation

To verify effectiveness of the proposed algorithm DFT, we conducted a computer experiment. For comparison, we adopted the fault-tolerant routing algorithm FT, which is proposed by Nishiyama et al.

We have implemented Algorithms FT and DFT, and executed a computer experiment following the procedure below.

Step 1) For each n = 6,7,8 of $HS(2n, n)$ and the ratio of faulty nodes f = 0, 0.04, ..., 0.20, repeat 10,000 times Steps 2 to 3.
Step 2) Select $\lfloor {}_{2n}C_n \times f \rfloor$ nodes from nodes in $HS(2n, n)$ randomly, and include them in the set of faulty nodes F.
Step 3) First, select the source node $\boldsymbol{s}$ from nonfaulty nodes randomly. Then, repeat 100 times Substeps 3-1 and 3-2.
Substep 3-1) Select the destination nodes randomly from the connected component that includes $\boldsymbol{s}$. Then, apply Algorithms FT and DFT to try to establish paths from $\boldsymbol{s}$ to $\boldsymbol{d}$ by calling FT ($\boldsymbol{s}$, $\boldsymbol{d}$, F) and DFT ($\boldsymbol{s}$, $\boldsymbol{d}$, F), respectively.
Substep 3-2) If a path from $\boldsymbol{s}$ to $\boldsymbol{d}$ is established, it is recorded as a successful routing.

After the experiment, we calculated the ratio of successful routings, which represents the ratio obtained by dividing the number of successful routings by the number of trials. Note that Algorithms FT and DFT always cause a routing failure by an infinite loop.

The results of the experiment are shown in Table II. The difference of the ratio of successful routings is at most 0.00684 which is attained by the ratio of faulty nodes about 12% in $HS(12, 6)$. With respect to the ratio of successful routings, the proposed algorithm DFT is superior to Algorithm FT regardless of the graph size and the ratio of faulty nodes. To recapitulate these results, Algorithm DFT attains a high ratio of successful routings, and it has higher performance compared to Algorithm FT.

Table II
Ratios of successful routings by Algorithms DFT and FT in $HS(2n, n)$

f	n = 6		n = 7		n = 8	
	FT	DFT	FT	DFT	FT	DFT
0.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.04	0.9991	0.9992	0.9981	0.9966	0.9967	0.9970
0.08	0.9855	0.9863	0.9801	0.9839	0.9764	0.9794
0.12	0.9547	0.9615	0.9473	0.9513	0.9432	0.9448
0.16	0.9134	0.9193	0.9039	0.9059	0.8970	0.8974
0.20	0.8635	0.8668	0.8504	0.8509	0.8447	0.8447

For complexities at each node, our algorithm takes $O(n^2)$ time to calculate all safety levels and the safety levels are exchanged with its neighbors $O(n^2)$ times in total. These values are same as those of the algorithms by Wu [10], Chiu and Chen [9], and Kaneko and Ito [11] for hypercube networks.

V. Conclusion

In this paper we have proposed a new fault-tolerant routing algorithm in a hyper-star graph $HS(2n, n)$ by introducing the notion of the directed safety levels. The time complexity of our method is $O(n^2)$, and its high reachability is verified from a computer experiment.

As a future work, it is interesting to apply our approach to other topologies for interconnection networks. Some fault-tolerant routing algorithms in hypercube networks use stochastic approaches [15], [16], [17], [18]. Hence, introducing stochastic structure into our routing algorithm is also included in the future works.

ACKNOWLEDGMENT

This study is partly supported by a Grant-in-Aid for Scientific Research (C) of the Japan Society for the Promotion of Science (JSPS) under Grant No. 22500041.

REFERENCES

[1] S. B. Akers and B. Krishnamurthy, "A group-theoretic model for symmetric interconnection networks," *IEEE Transactions on Computers*, vol. 38, pp. 555–566, Apr. 1989.

[2] P. F. Corbett, "Rotator graphs: An efficient topology for point-to-point multiprocessor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 3, pp. 622–626, May 1992.

[3] J.-S. Jwo, "Properties of star graph, bubble-sort graph, prefixreversal graph and complete-transposition graph," *Journal of Information Science and Engineering*, vol. 12, pp. 603–617, Dec. 1996.

[4] S. Latifi and P. K. Srimani, "A new fixed degree regular network for parallel processing," in *Proceedings of the Eighth IEEE Symposium on Parallel and Distributed Processing*, pp. 152–159, Oct. 1996.

[5] F. P. Preparata and J. Vuillemin, "The cube-connected cycles: A versatile network for parallel computation," *Communications of ACM*, vol. 24, pp. 300–309, May 1981.

[6] C. L. Seitz, "The cosmic cube," *Communications of the ACM*, vol. 28, pp. 22–33, Jan. 1985.

[7] H.-O. Lee, J.-S. Kim, E. Oh, and H.-S. Lim, "Hyper-star graph: A new interconnection network improving the network cost of the hypercube," in *Proceedings of the First EurAsian Conference on Information and Communication Technology*, pp. 858–865, Oct. 2002.

[8] G.-M. Chiu and S.-P. Wu, "A fault-tolerant routing strategy in hypercube multicomputers," *IEEE Transactions on Computers*, vol. 45, pp. 143–155, Feb. 1996.

[9] G.-M. Chiu and K.-S. Chen, "Use of routing capability for fault-tolerant routing in hypercube multicomputers," *IEEE Transactions on Computers*, vol. 46, pp. 953–958, Aug. 1997.

[10] J. Wu, "Adaptive fault-tolerant routing in cube-based multicomputers using safety vectors," *IEEE Transactions on* Parallel and Distributed Systems, vol. 9, pp. 322–334, Apr. 1998.

[11] K. Kaneko and H. Ito, "Fault-tolerant routing algorithms for hypercube interconnection networks," *IEICE Transactions on Information and Systems*, vol. E84-D, pp. 121–128, Jan. 2001.

[12] C.-B. Yang, K.-S. Huang, and C.-T. Tseng, "A fault-tolerant routing algorithm with safety vectors on the (n,k)-star graph," in *Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms, and Networks*, pp. 34–39, Dec. 2009.

[13] S.-I. Yeh, C.-B. Yang, and H.-C. Chen, "Fault-tolerant routing on the star graph with safety vectors," in *Proceedings of the Sixth Annual International Symposium on Parallel Architectures, Algorithms, and Networks*, pp. 301–306, May 2002.

[14] Y. Nishiyama, Y. Hirai, and K. Kaneko, "Fault-tolerant routing based on safety levels in a hyper-star graph," in *Proceedings of IADIS Applied Computing 2012*, pp. 348–352, Oct. 2012.

[15] J. Al-Sadi, K. Day, and M. Ould-Khaoua, "Fault-tolerant routing in hypercubes using probability vectors," *Parallel Computing*, vol. 27, pp. 1381–1399, Sept. 2001.

[16] D. T. Duong and K. Kaneko, "Fault-tolerant routing algorithms based on approximate routable probabilities for hypercube networks," in *Proceedings of the 2011 International Conference on Parallel and Distributed Processing Techniques and Applications*, vol. 1, pp. 116–122, 2011.

[17] D. T. Duong and K. Kaneko, "Fault-tolerant routing algorithms based on approximate directed routable probabilities for hypercubes," in *Proceedings of the 11th International Conference on Algorithms and Architectures for Parallel Processing*, pp. 106–116, Oct. 2011.

[18] M. Manabu and K. Kaneko, "A fault-tolerant routing algorithm using directed probabilities in hypercube networks," in *Proceedings of the 2012 International Conference on Parallel and Distributed Processing Techniques and Applications*, vol. 1, pp. 131–136, 2012.