Stochastic Fault-Tolerant Routing in Dual-Cubes

Junsuk PARK\(^{1}\), Nobuhiro SEKI\(^{1}\), Nonmembers, and Keiichi KANEKO\(^{1}(a)\), Member

SUMMARY In the topologies for interconnected nodes, it is desirable to have a low degree and a small diameter. For the same number of nodes, a dual-cube topology has almost half the degree compared to a hypercube while increasing the diameter by just one. Hence, it is a promising topology for interconnection networks of massively parallel systems. We propose here a stochastic fault-tolerant routing algorithm to find a non-faulty path from a source node to a destination node in a dual-cube.

key words: faulty edges, hypercube, multicomputer, interconnection network, parallel processing, dependable computing

1. Introduction

Recently, there have been many studies on parallel systems, and they have proposed several topologies for their interconnection networks. However, a state-of-the-art massively parallel system interconnects tens of thousands of nodes, so new topologies are necessary. In this study, we focus on such a topology, dual-cube [1]. The dual-cube is an extension of the hypercube [2], and it addresses some of the drawbacks of the hypercube when applied to massively parallel systems. Since the existence of faulty elements is inevitable in a massively parallel system, we propose here a fault-tolerant routing algorithm for dual-cubes.

The rest of this paper is organized as follows. Section 2 introduces the definitions of hypercube and dual-cube. Section 3 describes our algorithm in detail. Section 4 reports the results of a validation experiment, and Sect. 5 concludes this paper.

2. Preliminaries

We present here definitions of hypercube and dual-cube.

Definition 1: An \( n \)-hypercube \( Q_n \) consists of a set of nodes \( V(Q_n) \) and a set of edges \( E(Q_n) \) where \( V(Q_n) = \{0, 1\}^n \) and \( E(Q_n) = \{(a, b) | a, b \in V(Q_n), H(a, b) = 1\} \). Note that \( H(a, b) \) represents the Hamming distance between two bit sequences \( a \) and \( b \).

Definition 2: An \( n \)-dual-cube \( D_n \) consists of a set of nodes \( V(D_n) \) and a set of edges \( E(D_n) \) where \( V(D_n) = \{0, 1\}^{2n-1} \). For a node \( a = (a_1, a_2, \ldots, a_{2n-1}) \in V(D_n) \), if \( a_1 = 0 \), \( (a_2, \ldots, a_n) \) and \( (a_{n+1}, \ldots, a_{2n-1}) \) are called the cluster ID and the node ID, and denoted by \( C(a) \) and \( N(a) \), respectively. If \( a_1 = 1 \), \( (a_2, \ldots, a_n) \) and \( (a_{n+1}, \ldots, a_{2n-1}) \) are called the node ID and the cluster ID, and denoted by \( N(a) \) and \( C(a) \), respectively. For two nodes \( a = (a_1, a_2, \ldots, a_{2n-1}), b = (b_1, b_2, \ldots, b_{2n-1}) \in V(D_n) \), \( (a, b) \in E(D_n) \) if and only if either one of the conditions is satisfied:

\[
\begin{align*}
H(N(a), N(b)) &= 0, H(C(a), C(b)) = 0 \quad (a_1 \neq b_1), \\
H(N(a), N(b)) &= 1, H(C(a), C(b)) = 0 \quad (a_1 = b_1).
\end{align*}
\]

The number of nodes, the degree, and the diameter of \( D_n \) are equal to \( 2^{2n-1}, n \), and \( 2n \), respectively. On the other hand, the degree and the diameter of \( Q_{2n-1} \), which has the same number of nodes as both \( 2n - 1 \). Therefore, compared to \( Q_{2n-1} \), \( D_n \) can interconnect the same number of nodes with almost half the degree while increasing the diameter by just one.

Figure 1 illustrates \( Q_5 \) and \( D_3 \). Note that we can obtain \( D_1 \) by deleting some edges from \( Q_5 \). In general, by deleting some edges from \( Q_{2n-1} \), we can obtain \( D_n \).

3. Fault-Tolerant Routing

Because we can obtain \( D_n \) by deleting some edges from \( Q_{2n-1} \), we first pick up the stochastic edge-fault-tolerant routing algorithm FT1 for a hypercube, which was proposed by Lam et al. [3], to apply to a dual-cube. Moreover, we extend FT1 based on the concept of directed routable probability by Duong et al. [4] to invent an improved stochastic edge-fault-tolerant routing algorithm FT2.

Now, we introduce a function \( \gamma \) such that for an edge \( e, \gamma(e) = 0 \) if \( e \in F \) and \( \gamma(e) = 1 \) if \( e \notin F \) where \( F \) represents a set of faulty edges in \( Q_n \).

First, we explain Algorithm FT1 by Lam et al. In preprocessing of FT1, each node \( a \) calculates routing probability \( P_h(a) \) regarding Hamming distance \( h \). \( P_h(a) \) indicates an
approximate probability that \( a \) can send a message to any node \( d \) with \( H(a, d) = h \). \( P_h(a) \) can be recursively calculated by repeatedly exchanging the probabilities with neighbor nodes as follows:

\[
P_h(a) = \begin{cases} 
1 & (h = 0), \\
\sum_{n \in N(a)} \max_{m \in I} \gamma(a, n)P_{h-1}(n)/nC_h & (h > 0)
\end{cases}
\]

where \( N(a) \) represents a set of neighbor nodes of \( a \). In the routing process, let us assume that a node \( a \) has received a message for a destination node \( d \) with \( H(a, d) = h \) from a neighbor node \( p \). Then, the node \( a \) finds \( n^* = \arg \max_{n \in N(a)} \gamma(a, n)P_{h-1}(n) \) among the set of neighbor nodes \( N_0(a, d) \) that are on the shortest paths to \( d \), and forwards the message to it if \( \gamma(a, n^*)P_{h-1}(n^*) > 0 \). If \( \gamma(a, n^*)P_{h-1}(n^*) = 0 \), \( a \) forwards the message to the node \( h = \arg \max_{n \in N(a)} \gamma(a, n)P_{h+1}(n) \) among the set of neighbor nodes \( N_1(a, d) \) on the detour paths to \( d \).

Next, we explain Algorithm FT2, which uses directed routing probabilities \( P'_h(a, n_0) \). In preprocessing, each node \( a \) calculates the probabilities by repeatedly exchanging them with its neighbor nodes similarly to FT1. However, when \( a \) exchanges its approximate probability with its neighbor node \( n_0 \), it excludes the effect by \( n_0 \) from the probability. \( P'_h(a, n_0) \) can be calculated recursively as follows:

\[
P'_h(a, n_0) = \begin{cases} 
1 & (h = 0), \\
\sum_{n \in N(a)} \max_{m \in I} \gamma(a, n)P'_{h-1}(n, a)/nC_h & (h > 0)
\end{cases}
\]

The routing process is same as that of FT1.

4. Performance Evaluation

To evaluate performance of Algorithms FT1 and FT2, we conducted a validation experiment. For the baseline, we adopted a node-fault-tolerant limited-safety-level-based routing algorithm JW proposed by Jiang and Wu [5]. Since an edge-fault-tolerant routing algorithm can tolerate faulty nodes by regarding all the edges incident to the faulty nodes as faulty, we assume only faulty nodes in the experiment. In \( D_0 \), for each ratio of faulty nodes \( \alpha \) (\( 0.0 \leq \alpha \leq 0.5 \)), we repeated the following steps 10,000 times for randomly-generated pairs of the source node \( s \) and the destination node \( d \).

1. Select \( \lceil \alpha 2^{11} \rceil \) faulty nodes randomly.
2. Select the source node \( s \) among the non-faulty nodes randomly.
3. Select the destination node \( d \) among the non-faulty nodes other than \( s \) randomly. If there is not any non-faulty path from \( s \) to \( d \), go back to Step 1.

4. For \( s \) and \( d \), apply Algorithms FT1, FT2, and JW, and count the number of successful routings.

Figure 2 shows the ratio of successful routings and the average path length. Though these values are in the trade-off relation, the former is much more important. Hence, from this figure, we can see that FT2 showed the best performance regarding successful routings among the three algorithms.

5. Conclusion

We have proposed here a stochastic edge-fault-tolerant routing algorithm for dual-cubes. The result of our validation experiment showed that our algorithm performed better than the algorithm proposed in earlier research.

Acknowledgments

We appreciate Dr. Bipin Indurkhya for proofreading this paper and the reviewer for his valuable comment. This study was partly supported by a Grant-in-Aid for Scientific Research (C) of the Japan Society for the Promotion of Science under Grant No. 17K00993.

References