Conditional Fault-Diameter of Torus Networks

Abderezak Touzene and Khaled Day

Department of Computer Science, College of Science, Sultan Qaboos University, P.O. Box 36, Al-Khod 123, Sultanate of Oman, Email: touzene@squ.edu.om, kday@squ.edu.om.

Torus

Torus   :

Torus

Torus   .

Torus   .

)   Torus   .

(star-graph   hypercube
.

.

ABSTRACT: We obtain the conditional fault-diameter of the square torus interconnection network under the condition of forbidden faulty sets (i.e. assuming that each non-faulty processor has at least one non-faulty neighbor). We show that under this condition, the square torus, whose connectivity is 4, can tolerate up to 5 faulty nodes without becoming disconnected. The conditional node connectivity is, therefore, 6. We also show that the conditional fault-diameter of the square torus is equal to the fault-free diameter plus two. With this result the torus joins a group of interconnection networks (including the hypercube and the star-graph) whose conditional fault-diameter has been shown to be only two units over the fault-free diameter. Two fault-tolerant routing algorithms are discussed based on the proposed vertex disjoint paths construction.

KEYWORDS: Fault-tolerance, forbidden faulty sets, torus, node connectivity, conditional node connectivity, conditional fault-diameter, fault-tolerant routing.

1. Introduction

The node connectivity and the fault diameter have been used as measures of the fault-tolerance of interconnection networks. These measures however do not reflect the real resilience of these networks. It is true that when the number of faulty nodes is equal to the connectivity, the network may become disconnected. However this is very unlikely to happen since only very special fault distributions of these faults cause disconnection. For instance, in the torus, the node-connectivity is 4 and therefore 4 faulty nodes may cause disconnection. However, the network becomes disconnected only when all the 4 faulty nodes are neighbors of the same node which is very improbable.

In an attempt to better quantify the fault resilience of a network, the concept of forbidden faulty sets has been introduced Esfahanian (1989). The idea is to assume that each node has at least one non-faulty neighbor. Under this forbidden faulty set condition the number of tolerable faulty nodes is significantly larger with a slight increase in the fault diameter. Esfahanian (1989) proved that for the binary n-cube, whose connectivity is n, $2n$-3 nodes can fail (under the forbidden faulty set condition) without disconnecting the network. Latifi (1993) then showed that the corresponding conditional fault diameter increases only by 2. In Rouskov *et al.* (1996) similar results for the star graph network have been established. (Latifi and Naraghi-Pour, 1994) has generalized this idea by assuming that each node has at least k non-faulty neighbors. Similar results for the m-ary generalized n-cube network have also been obtained (Wu, 1998).

In this paper we contribute to the study of the properties of the torus by establishing its conditional node connectivity and its conditional fault-diameter. We show that under the condition of forbidden faulty sets, the torus (whose connectivity is 4) can tolerate up to 5 faulty nodes without becoming disconnected. The corresponding conditional fault-diameter is shown to be equal to the fault-free diameter plus two (i.e. $k + 2$). A fault tolerant routing algorithm is discussed based on the construction of node-disjoint paths in the presence of faults under the forbidden faulty sets condition. The rest of the paper is organized as follows. In section 2 we introduce the notations and include some preliminaries. Section 3 establishes the conditional faulty diameter of the torus. In section 4 we discuss how a fault tolerant routing can be derived and section 5 concludes the paper.

2. Notations and Preliminaries

***Definition* 1:** The node-connectivity $C(G)$ (or point-connectivity) of a graph G is the minimum number of nodes of G whose removal results in a disconnected or trivial graph.

For any two nodes in the torus there are four node-disjoint paths connecting them (Day and Al-Ayyoub, 1997), therefore $C(torus) = 4$. We now define the conditional node connectivity.

***Definition* 2:** The conditional node connectivity $CC(G)$ of a graph G is the minimum number of nodes of G whose removal results in a disconnected or trivial graph, provided that each of the remaining nodes has at least one adjacent node in G that is not removed.

We will obtain the conditional node connectivity of the torus as a consequence of obtaining its conditional fault-diameter.

***Definition* 3:** The square torus (k-torus) is a wrap-around mesh which consists of k rows and k columns. Each row and each column consists of a cycle of k nodes. Any two nodes are connected by two paths one is called the shortest path on the cycle and the other is called the longest path on the cycle.

In what follows we will adopt the following notations: If we consider two nodes x and y in the same column C^m (x^m, and y^m), $m=0..k-1$, we denote by x_S^m, y_S^m the respective neighbors of x and y located on the shortest path between x and y in C^m. We denote by x_L^m, y_L^m the respective neighbors of x and y located on the longest path between x and y in C^m. We use similar notation for the second

node in the shortest path $x_{SS}{}^m$, $y_{SS}{}^m$. And $x_{LL}{}^m$, $y_{LL}{}^m$ are the second nodes in the longest path between x and y in C^m. Any node x^m in C^m has 4 neighbors: $x_S{}^m$, $x_L{}^m$, x^{m+1}, x^{m-1}. See Figure 1.
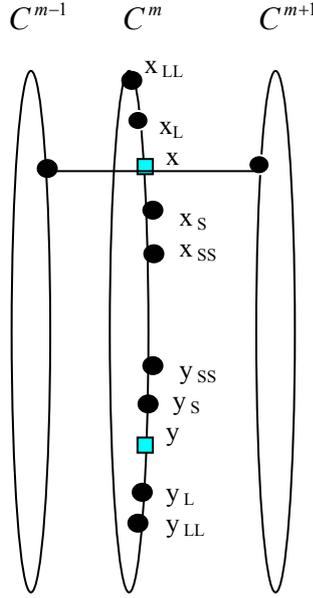

Figure 1. x and y in the same cycle C^m.

3. Conditional Fault-Diameter of the *k*-Torus

The fault-diameter of an interconnection network is the maximum distance between any two nodes in the presence of a number of faults equal to the connectivity of the network minus one.
***Definition* 4:** The fault-diameter $FD(G)$ of a graph G is the maximum distance between any two nodes of G in the presence of at most $C(G)$-1 faulty nodes.
***Definition* 5:** The conditional node diameter $CFD(G)$ of a graph G is the maximum distance between any two nodes of G in the presence of at most $C(G)$-1 faulty nodes, provided that each of the non-faulty nodes has at least one non-faulty adjacent node in G.
We now obtain the conditional fault-diameter of the *k-torus*.
***Theorem* 1.** The conditional fault diameter of the *k-torus* under the forbidden sets assumption is $k+2$.
Proof: Consider two nodes x and y at distance $D(x, y)=L$ in a *k-torus* with at most 5 faulty nodes. In the first case, we consider that x and y belong to the same column C^0. The case where x and y belong to a same row is symmetric and can be treated similarly. In the second case we consider that x and y are located in different columns and different rows of the torus.
Case 1. If x and y are on the same columns (say C^0) and similarly for x and y on the same row.
Case 1.1. If all faults are either outside the target cycle C^0 or are located along the longest path between x and y, then the shortest path between x and y is fault-free of length $L \leq k+2$
Case 1.2. If there is only one fault in the target cycle and this fault is located on the shortest path between x and y, then the longest path between x and y is fault-free of length $k-L \leq k+2$.
1.3. If the number of faults in the target cycle is 4 or 5, then there are two node-disjoint paths between x and y:
- $B1$: x, x^1, shortest path in C^1 to reach y^1, y.

- $B2$: x, x^{k-1}, shortest path in C^{k-1} to reach y^{k-1}, y.

At least one of these two paths is fault-free. The length of these paths is $L+2 \leq k+2$. See Figure 2 (a).

Case 1.4. The number of faults in C^0 is 2 or 3:

We discuss several cases based on the number of faults which are neighbors of x and y.

Case 1.4.1. If these faults are located on the shortest path between x and y, then we have the same conclusion as in case 1.2.

Case 1.4.2. All faults are not neighbors of x or y. The following four paths have no common nodes other than x, y and their neighbors:

- $P1$: x, x_S^0, x_S^1, shortest path in C^1 to reach y_s^1, y_s, y.
- $P2$: x, x_S^0, x_S^{k-1}, shortest path in C^{k-1} to reach y_S^{k-1}, y_s, y.
- $P3$: x, x_L^0, x_L^1, longest path in C^1 to reach y_L^1, y_L, y.
- $P4$: x, x_L^0, x_L^{k-1}, longest path in C^{k-1} to reach y_L^{k-1}, y_L, y.

If the number of faults in C^0 is 3, then at least two of these paths are fault-free.
If the number of faults in C^0 is 2, then at least one of these paths is fault-free.
The length of the above paths is less than $k+1 < k+2$. See Figure 2 (b).

Case 1.4.3. One fault is neighbor of y, the second and the third (if 3 faults) are not a neighbor of x: this case can be solved using slight modification of the paths in case 1.4.2 as follows:
If the faulty neighbor of y is y_S^0: We use $P3$, $P4$ and

- $M1$: x, x_S^0, x_S^1, shortest path in C^1 to reach y^1, y (path length $L+2 \leq k+2$).
- $M2$: x, x_S^0, x_S^{k-1}, shortest path in C^{k-1} to reach y^{k-1}, y (path length $L+2 \leq k+2$).

If the faulty neighbor of y is y_L^0, we use $P1$, $P2$ and

- $M3$: x, x_L^0, x_L^1, longest path in C^1 to reach y_L^1, y^1, y (path length $k-L+2 \leq k+2$)
- $M4$: x, x_L^0, x_L^{k-1}, longest path in C^{k-1} to reach y_L^{k-1}, y^{k-1}, y (path length $k-L+2$ $k+2$). See Figure 2 (c).

The symmetric case where the fault is neighbor of x is similar.

Case 1.4.4. Two faults are neighbors of y (y_S^0 and y_L^0), the third fault, if any, is not neighbor of x. There are four paths: $M1$, $M2$, and

- $S1$: x, x^1, x^2, shortest path in C^2 to reach y^2, y^1, y (path length $L+4 \leq k+2$).
- $S2$: x, x^{k-1}, x^{k-2}, shortest path in C^{k-2} to reach y^{k-2}, y^{k-1}, y (path length $L+4 \leq k+2$).

Note that y^1 and y^{k-1} cannot be both faulty at the same time (forbidden sets assumption). If both are not faulty, then one of these four paths will be fault-free. Assume now that one of them is faulty (say y^{k-1}). We end up only with $M1$, $S1$ (a minimum of three faults already consumed). An additional path is $M3$. In the presence of a maximum of 2 other faults, one among the three paths is fault-free of length less than or equal to $k-L+2 < k+2$. See Figure 2 and 3. Now we consider a third fault in C^0, which is a neighbor of x. If the third faulty node is x_L^0, then we make use of the same paths even without the need to $M3$. If the third faulty node is x_S^0, then we use the four paths: $B1$, $B2$, $M3$ and $M4$. Since y^1 and y^{k-1} are common nodes to the four paths, the same distribution of faults on y^1 and y^{k-1} leads to the same conclusions.

The case where both faults are neighbors of x only is a symmetric case and can be solved similarly.

Case 1.4.5. One fault is neighbor of y and the second is neighbor of x: If the faulty neighbor of x is x_L^0 we will use the two paths $P1$, $P2$ (if y_S^0 is fault-free), or $M1$ and $M2$ (if y_S^0 is faulty). The two other paths are:

- $M5$: x, x^1, longest path in C^1 to reach y_L^1, y^1, y (path length $k-L+2 \leq k+2$),
- $M6$: x, x^{k-1}, longest path in C^{k-1} to reach y_L^{k-1}, y^{k-1}, y (path length $k-L+2 \leq k+2$), if y_L^0 is faulty (y_S^0 is fault-free), or
- $M7$: x, x^1, longest path in C^1 to reach y_L^1, y_L, y (path length $L+2 \leq k+2$),
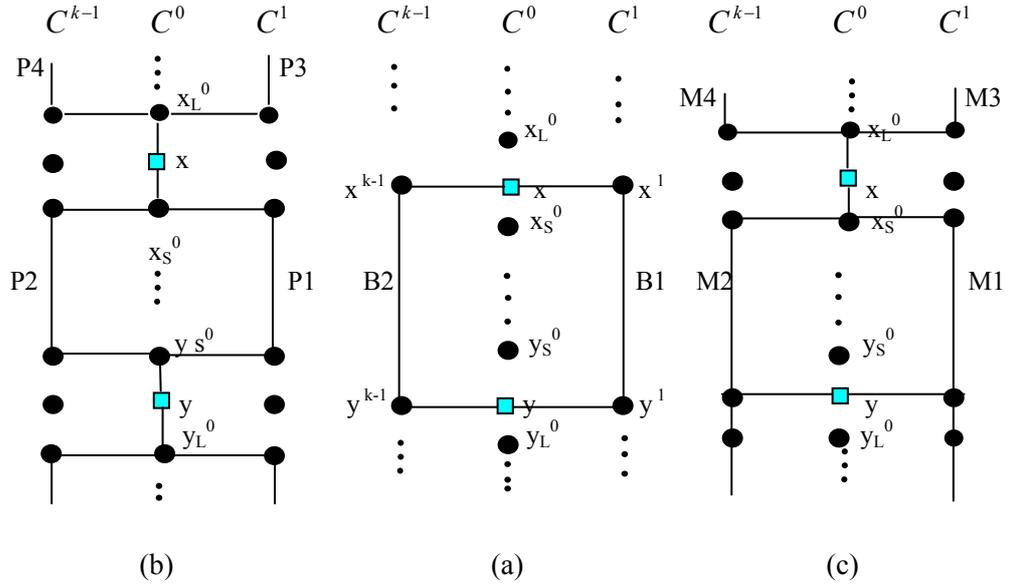
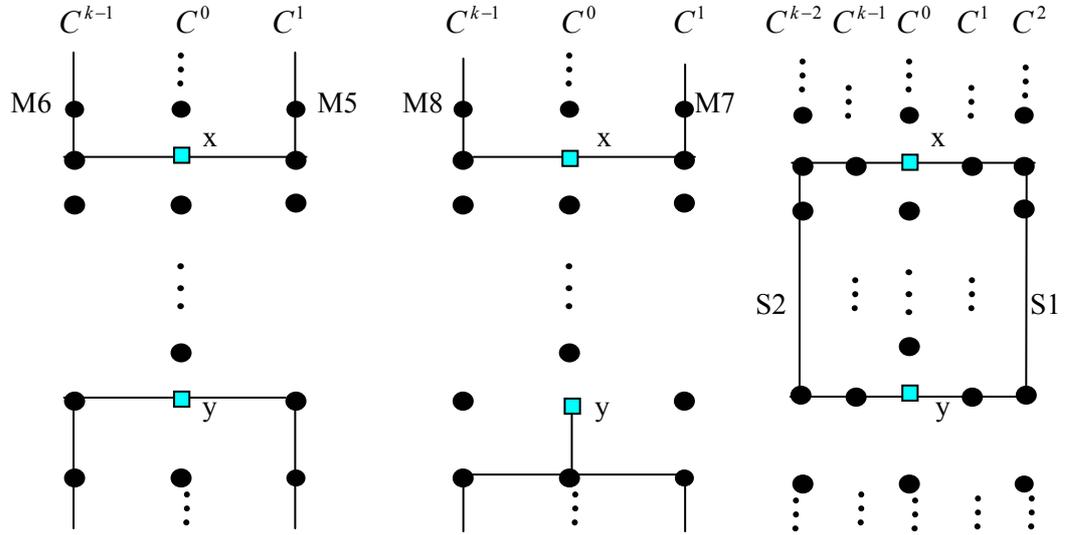Figure 2. Paths for cases 1.3, 1.4.2 and 1.4.3.

Figure 3. Paths for cases 1.4.3, 1.4.4 and 1.4.5.

- *M8*: x, x^{k-1}, longest path in C^{k-1} to reach y_L^{k-1}, y_L, y (path length $L+2 \leq k+2$), if y_L^0 is fault-free (y_S^0 is faulty). Now if the neighbor of x, x_S^0 is faulty: A similar technique is used to construct four node-disjoint paths (the common nodes are fault-free) between x and y. These paths will depend on which neighbor of y will be faulty (y_S^0 or y_L^0).
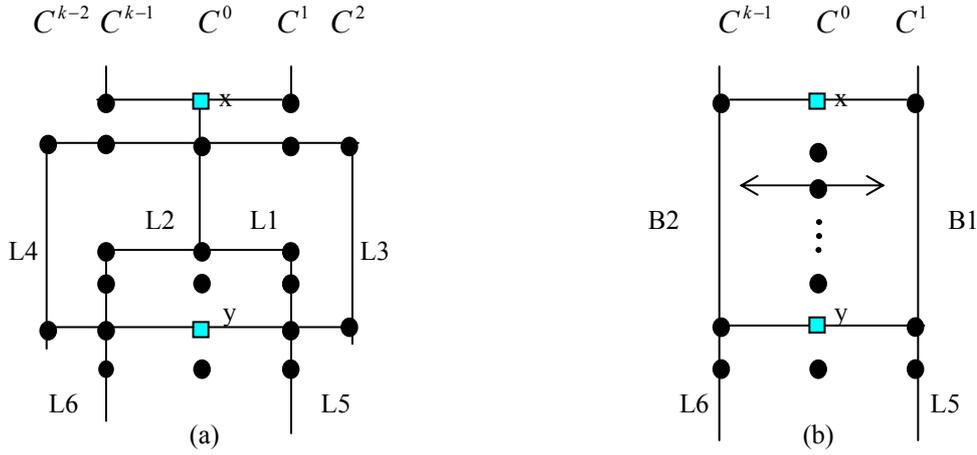
Figure 4. Paths for case 1.4.6.

Case 1.4.6. The number of faults in C^0 is two and both faults are neighbors of x and y at the same time: In this case y^0_S, y^0_L are faulty and $x^0_L = y^0_L$, $(L=2)$. The case $x^0_S = y^0_S$ is symmetric and can be solved similarly. We can construct 6 paths as follows:

- $L1$: x, x^0_S, shortest path in C^0 to reach y^0_{SS}, $y_{SS}{}^1$, $y_S{}^1$, y^1, y (path length $L+2 \le k+2$).
- $L2$: x, x^0_S, shortest path in C^0 to reach y^0_{SS}, $y_{SS}{}^{k-1}$, $y_S{}^{k-1}$, y^{k-1}, y (path length $L+2 \le k+2$).
- $L3$: x, x^0_S, $x_S{}^1$, $x_S{}^2$, shortest path in C^2 to reach y^2, y^1, y (path length $L+4 \le k+2$).
- $L4$: x, x^0_S, $x_S{}^{k-1}$, $x_S{}^{k-2}$, shortest path in C^{k-2} to reach y^{k-2}, y^{k-1}, y (path length $L+4 \le k+2$).
- $L5$: x, x^1, $y_L{}^1$, y^1, y (path length 4).
- L6: x, x^{k-1}, $y_L{}^{k-1}$, y^{k-1}, y (path length 4).

Note that the nodes y^1 and y^{k-1} are common nodes to the six paths, but fortunately they cannot be faulty at the same time (forbidden sets assumption). See Figure 4 (a). If both are not faulty, then in presence of 3 faults outside C^0, three among the six above paths will be fault-free. If we assume that y^{k-1} is faulty and y^1 is fault-free, then at least one of the three paths $L1$, $L3$, $L5$ will be fault-free. If the node y^{k-1} is not faulty and y^1 is faulty, then at least one of the three paths $L2$, $L4$, $L6$ will be fault-free. The length of the above paths is less than or equal to $k+2$.

Now if we consider that there is a third faulty node in C^0, which is $x_S{}^0$. We make use of the following paths: $B1$, $B2$ and $L5$, $L6$. See Figure 4(b). Note that these paths have some common nodes: $L5$ and $B1$ have two nodes in common, which are x^1 and y^1. $L6$ and $B2$ have two nodes in common, which are x^{k-1} and y^{k-1}. In presence of two faults outside C^0, if all these four nodes are not faulty, then there are at least two fault-free paths between x and y. Under the forbidden sets assumption, nodes x^1 and x^{k-1} cannot be both faulty in the same time. And nodes y^1 and y^{k-1} cannot be both faulty in the same time. If both nodes x^1 and y^1 are faulty, then $B2$ and $L6$ will be fault-free.

If both nodes x^{k-1} and y^{k-1} are faulty, then $B1$ and $L5$ will be fault-free. If y^{k-1} and x^1 are faulty, then an alternative path will be x, x^{k-1} in $B2$, $x_S{}^{k-1}$ in $B2$, $x_{SS}{}^{k-1}$ in $B2$, $x_{SS}{}^0$ in C^0, $x_{SS}{}^1$ in $B1$, continue on $B1$ to reach y. And similarly for the symmetric case (y^1 and x^{k-1} are faulty), jump from $B1$ to $B2$ from the node $x_{SS}{}^1$ in $B1$ to the node $x_{SS}{}^0$, $x_{SS}{}^{k-1}$ in $B2$, then continue on $B2$. The length of these paths is less than or equal to $k+2$.

Case 2. Nodes x and y do not belong to the same column or the same row. Without loss of generality, we assume that $x = x^0$ belongs to C^0, and $y = y^m$ belongs to C^m. Let us consider the six-node disjoint paths connecting the columns C^0 to C^m. These paths can be divided into two groups:
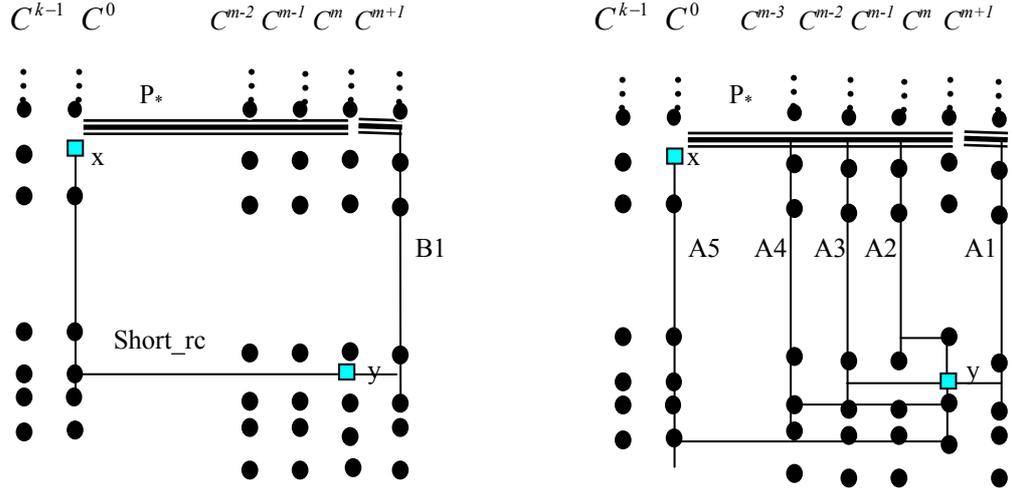
56

Figure 5. Paths for cases 2.3 and 2.4.1.

Group 1

- x_L^0, x_L^1, x_L^2 , shortest path in this row to reach x_L^{m-1} , x_L^m .
- x, x^1, x^2 , shortest path in this row to reach x^{m-1}, x^m .

Group 2

- x_S^0, x_S^1, x_S^2 , shortest path in this row to reach x_S^{m-1} , x_S^m .
- y_L^0, y_L^1, y_L^2 , shortest path in this row to reach y_L^{m-1} , y_L^m .
- y^0, y^1, y^2 , shortest path in this row to reach y^{m-1}, y^m .
- y_S^0, y_S^1, y_S^2 , shortest path in this row to reach y_S^{m-1}, y_S^m .

In presence of five faults, at least one of the above six paths is fault-free. If the fault-free path is in group 1, then we will construct paths starting from x going to y. Otherwise, we construct paths starting from y going to x. In what follows, to simplify the analysis, we will always consider that the fault-free path is from group 1, and we denote this path P_*. The other case is similar (symmetric).

Case 2.1. The column C^m is fault-free: There is at least one fault-free path from x to y, which starts at x, P_* to reach x_*^m, shortest path in C^m to reach y (x_*^m represents x_S^m, x^m or x_L^m depending on which of the three paths of group 1 corresponds to P_*). The length of the constructed fault-free path between x and y is at most $L+2 \leq k+2$.

Case 2.2. The column C^m contains 5 faults: The following path named *Short_rc* (rc stands for rows then columns): x, x_S^0, shortest path in C^0 to reach y^0, shortest path in this row to reach y^{m-1} , y. This path is fault-free of length $L \leq k+2$.

Case 2.3. The number of faults in C^m is 4. We use the path *Short_rc*, if it is fault-free. Otherwise *Short_rc* contains the fifth fault and there are no more faults outside $\bar{C}^m$. An alternative way is to start from x, P_* to reach x_*^m, x_*^{m+1} , shortest path on C^{m+1} to reach y^{m+1} , y. The length of the constructed fault-free path using $B1$ is $L+4$. See Figure 5.

Remark **1.** If $L \leq k$-2, then the length of the constructed path is less than or equal to $k+2$. Otherwise, ($L = k$-1, or k), we construct another path as follows: x, x^{k-1}, x^{k-2}, longest path on this row to reach x^{m+1} , shortest path in C^{m+1} to reach y^{m+1} , y. It is simple to see that the length of this fault-free path in all cases ($L > k$-2) is less than or equal to $k+2$.

57

Case 2.4. The number of faults in C^m is 1 (4 faults outside).

Case 2.4.1. The nodes $y_S{}^m$, $y_L{}^m$ are both not faulty: We consider the 5 following paths:

- $A1$: x, P_* to reach $x_*{}^m$, $x_*{}^{m+1}$ shortest path in C^{m+1} to reach y^{m+1}, y (path length $L+4$).
- $A2$: x, P_* to reach $x_*{}^{m-1}$ shortest path in C^{m-1} to reach $y_S{}^{m-1}$, $y_S{}^m$, y (path length $L+2$).
- $A3$: x, P_* to reach $x_*{}^{m-2}$ shortest path in C^{m-2} to reach y^{m-2}, y^{m-1}, y (path length $L+2$).
- $A4$: x, P_* to reach $x_*{}^{m-3}$ shortest path in C^{m-3} to reach $y_L{}^{m-3}$, $y_L{}^{m-2}$, $y_L{}^{m-1}$, $y_L{}^m$, y (path length $L+4$).
- $A5$: x, $x_S{}^0$, shortest path in C^0 to reach $y_{LL}{}^0$, shortest path in this row to reach $y_{LL}{}^m$, $y_L{}^m$, y (path length $L+4$).

These five paths are node-disjoint or the common nodes are fault-free. At least one among the five path will be fault-free. If the fault-free path is of length $L+4$, then if $L \leq k$-2, the considered path will have a length $\leq k+2$. Otherwise, we construct an alternative path using a similar technique as described in remark 1. See Figure 5.

***Remark* 2.** The existence of the paths $A4$, $A3$, $A2$ is subject to the condition: the column distance c between C^0 and C^m is greater or equal to 4. Otherwise, we consider the extreme case where no one of these paths exists (c=1). Since there is only one fault in C^m, if the fault belongs to the longest path between x^m and y in C^m, then the path x, x^m, shortest path in C^m to reach y, will be fault-free. Otherwise, the longest path between x^m and y in C^m will be fault-free and we construct five paths as follows:

- 1: x, $x_L{}^0$, P_L to reach $x_L{}^m$, $x_L{}^{m+1}$, shortest path in C^{m+1} to reach $y_S{}^{m+1}$, $y_S{}^m$, y.
- 2: x, $x_L{}^0$, P_L to reach $x_L{}^m$, longest path in C^m to reach y.
- 3: x, $x_L{}^0$, P_L to reach $x_L{}^m$, $x_{LL}{}^m$, $x_{LL}{}^{m+1}$, longest path in C^{m+1} to reach y^{m+1}, y.
- 4: x, $x_S{}^0$, shortest path in C^0 to reach y^0, shortest path in this row to reach y.
- 5: x, $x_L{}^0$, longest path in C^0 to reach $y_L{}^0$, shortest path in this row to reach y.

These five paths are node disjoint or the common nodes are fault-free. If we denote by r the row distance between x and y, the longest path will be of length k-r+1 $\leq$ k+2. See Figure 6.

Case 2.4.2. Node $y_S{}^m$ is faulty. The case where $y_L{}^m$ is faulty is trivial, because the shortest path between $x_*{}^m$ and y will be fault-free and of maximum length of $k+2$.

We construct the following four paths:

- $C1$: x, P_* to reach $x_*{}^m$, $x_*{}^{m+1}$, shortest path in C^{m+1} to reach y^{m+1}, y (path length $L+4$).
- $C2$: x, P_* to reach $x_*{}^{m-1}$ shortest path in C^{m-1} to reach y^{m-1}, y (path length $L+2$).
- $C3$: x, P_* to reach $x_*{}^{m-2}$ shortest path in C^{m-2} to reach $y_L{}^{m-2}$, $y_L{}^{m-1}$, $y_L{}^m$, y (path length $L+4$).
- $C4$: x, $x_S{}^0$, shortest path in C^0 to reach $y_{LL}{}^0$, shortest path in this row to reach $y_{LL}{}^m$, $y_L{}^m$, y (path length $L+4$).

If any one of the above paths is fault-free, the problem is solved. Otherwise, all the faults are consumed. We focus on the fault of path $C4$: If the fault belongs to the shortest path within C^0, an alternative path (disjoint with $C1$, $C2$, $C3$) will be x, x^1, shortest path in C^1 to reach $y_{LL}{}^1$, shortest path in this row to reach $y_{LL}{}^m$, $y_L{}^m$, y (path length $\leq L+4$). If the faulty node is not in C^0, an alternative path will be *: $C4'$:* x, x_S, shortest path in C^0 to reach $y_{LL}{}^0$, $y_{LLL}{}^0$ (successor of $y_{LL}{}^0$), shortest path in this row to reach $y_{LLL}{}^m$, $y_{LL}{}^m$, $y_L{}^m$, y (path length $L+6$). If $L \leq k$-4, this path is fault-free of length $\leq k+2$. See Figure 7 (a).

***Remark* 3.** If $L > k$-4, we need to construct another path using longest path which are not much longer than the shortest paths (see remark 1 for $L= k$ or k-s). Let us decompose $L= r+c$, where r (respectively c) is the row distance (respectively the column distance) between x and y. If we focus on the case $L= k$-3, there are two cases:

- $r= \lfloor k/2 \rfloor$ and $c=\lfloor k/2 \rfloor - 3$, the alternative path will be Pr: x, $x_L{}^0$, longest path in C^0 to reach $y_{LLL}{}^0$, shortest path in this row to reach $y_{LLL}{}^m$, $y_{LL}{}^m$, $y_L{}^m$, y. (path length=k-2 $< k+2$). If

$c = \lfloor k/2 \rfloor$ and $r = \lfloor k/2 \rfloor - 3$, the alternative path will be Pc: x, x^{k-1}, longest path that row to reach x^{m+2}, shortest path in C^{m+2} to reach y_L^{m+2}, y_L^{m+1}, y_L^m, y. (path length=k-2 < k+2)

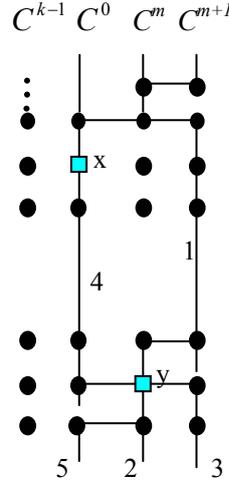$$C^{k-1} \; C^0 \quad C^m \; C^{m+1}$$

Figure 6. Distance between C^0 and C^m is 1 .

- $r = \lfloor k/2 \rfloor - 1$ and $c = \lfloor k/2 \rfloor - 2$, the alternative path is Pr (path length= k-1 < k+2). Pc will be used for the reversed case.

A similar technique leads to the construction of a fault-free path between x and y, of length $\leq k+2$, when $L = k$-2.

Case 2.5. The number of faults in C^m is 2 or 3:

Case 2.5.1. The nodes y_S^m, y_L^m are both not faulty or only one of them is faulty: Similar to **2.4.**

Case 2.5.2. The nodes y_S^m, y_L^m are both faulty: The nodes y^{m-1}, y^{m+1} could not be both faulty at the same time (forbidden sets assumption).

Case 2.5.2.1. If the node y^{m-1} is not faulty. We consider the paths $C1$, $C2$, $A3$ and

- $D1$: x, x_S, shortest path in C^0 to reach y_L^0, shortest path in this row to reach y_L^{m-1}, y^{m-1}, y (path length $L+2$).

Note that the three paths $A3$, $C2$, $D1$ have only one common node (y^{m-1}). If there are 3 faults in C^m, then two among the four above paths will be fault-free.

If there are only two faults in C^m, then only one among the four above paths will be fault-free.

Case 2.5.2.2. If the node y^{m+1} is fault-free (y^{m-1} is faulty), we consider the paths $C1$ and

- $D2$: x, x_S, shortest path in C^0 to reach y_{LL}^0, shortest path in this row to reach y_{LL}^{m+1}, y_L^{m+1}, y^{m+1}, y. (path length $L+6$)

If $C1$ is fault-free, the problem is solved. Otherwise, we use the following path:

- $D3$: is a modification of the path $C1$, bridging the faulty node, using the nodes of C^{m+2}, then back to $C1$ (path length $L+6$).

In the presence of three faults in C^m, $D2$ and $D3$ will be fault-free. If there are only two faults in C^m, then only $D2$ or $D3$ will be fault-free. The longest path ($L+6$) will be acceptable when $L \leq k$-4. Otherwise, an alternative path of length less than $k+2$ will be derived using the same technique as in remark 3. See Figure 7 (b).
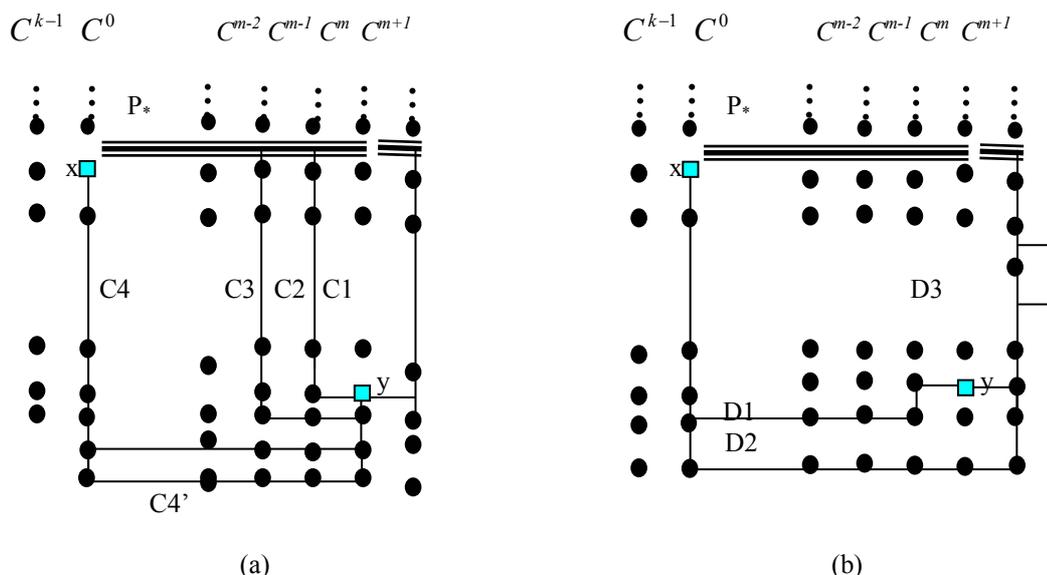
C^{k-1} C^0 C^{m-2} C^{m-1} C^m C^{m+1} C^{k-1} C^0 C^{m-2} C^{m-1} C^m C^{m+1}

P_* P_*

x   x

C4   C3 C2 C1   D3

y   D1   y

D2

C4'

(a)   (b)

Figure 7. Paths for cases 2.4.2 and 2.5.

4.   Fault-Tolerant Routing

We now illustrate the usefulness of the fault-free paths constructions of the previous section by describing two fault-tolerant routing algorithms that make use of these paths. Fault-tolerant routing has drawn a lot of attention in the literature (Suh *et al.* 2000; Chen *et al.* 2001; Ho *et al.* 2002; Wu 2003). Different ways of achieving fault-tolerant routing have been proposed. Some methods require extra hardware resources to route packets around faulty components while others simply bypass faulty components, as well as some healthy components, to maintain network regularity such as in (IBM 2002). Another technique is based on reconfiguring the routing tables in the case of failure, adapting them to the new topology after the failure such as in Casado *et al.* (2001), Pinkston *et al.* (2003) and Lysne *et al.* (2003). A more recently proposed methodology uses intermediate nodes (Gomez *et al.* 2004). The methodology is valid for any network topology. In Nordbotten *et al.* (2004), the idea of using intermediate nodes is extended to using multiple intermediate nodes for some paths.

In what follows, we describe two fault-tolerant routing algorithms for the square torus. It has been shown in the previous section that for any two nodes *X* and *Y* at distance *L*, we are able to construct between these two nodes a fault-free path of length at most $L+6 \leq k+2$ (assuming at most five faulty nodes satisfying the forbidden faulty set condition). We propose the following two fault-tolerant routing algorithms:

4.1 Parallel Routing Algorithm

Parallel routing algorithm has been used to design fault tolerant routing for the star graph in Rouskov *et al.* (1996). In our case, the basic idea is first to identify the case (see section 3) according to the location of *X* and *Y* in the torus and then apply the described construction of the parallel paths between *X* and *Y*. The routing process can send a message in parallel on the constructed vertex-disjoint paths to guarantee delivery because at least one of these paths is not faulty. This routing process is known as parallel routing algorithm. It consists of generating all the possible paths of interesting length between the source and the destination nodes. The generation of the routes results

from the union of all the paths generated for each case described in the previous section. As an example, if the source node *X* and the destination node *Y* do not belong to the same row and not to the same column, all the paths of case two will be generated. The generation of all paths of the case is necessary because the distribution of the faults is not known. Each fault distribution may correspond to a sub-case of the given case and thus to a set of paths to cover the corresponding fault distribution. The disadvantage of this routing algorithm is the communication overhead due to the presence of multiple copies of the same message in the network. If a wormhole routing model is used, this algorithm may increase link contention problem and thus the communication delay.

4.2 Single Path Fault-Tolerant Routing Algorithm

It is possible to achieve lower cost fault-tolerant routing using a single routing path but at the expense of additional failure detection and routing path reconfiguration mechanisms. A similar approach has been used in Day and Al-Ayyoub (2001) for the star graph. We can associate with each source node *X* and destination node *Y*, at any time, only one path among all the possible paths generated in the cases of the previous section. This path is called the active routing path. In our routing algorithm the choice of the active routing path starts by simply selecting the shortest path among the generated ones (cases) to ensure high communication efficiency. In fact, this routing approach requires link failure detection mechanisms and a mechanism for switching to a new active routing path for a given source node when a link failure is detected on its current active routing path. First we make the following failure assumptions:

1. Lower level fault detection and diagnosis mechanisms exist. Each node knows exactly the failure status of all its adjacent links. Each node's information about the status of its adjacent links is updated dynamically via these fault detection and diagnosis mechanisms.
2. Link failure and repairs may happen at any time, but the total number of faulty links does not exceed 5 under forbidden faulty set condition at any time.
3. We also assume that the delay for detecting an adjacent link failure is negligible.

A routing initiated at a source node *X* continues to follow the edges of its current active routing path (ARP), until a link failure is detected. The node that has detected the failure will report it to the source node *X* using the ARP backward. The source node will switch to another ARP chosen from the remaining possible paths (the shortest among them) and then resends the message using the new ARP.

This approach guarantees message delivery; in the worst case it may use all the possible ARPs (at least one among them is fault-free). This approach is efficient because it uses shortest paths as much as possible and it reduces considerable link contentions under heavy load network condition.

5. Conclusion

We have contributed in this paper to the study of the fault-tolerance of the square torus interconnection network by establishing its conditional fault-diameter under the condition of forbidden faulty sets (i.e., assuming that each non-faulty processor has at least one non-faulty neighbor). We have shown that under this condition the *k*-torus, whose connectivity is 4, can tolerate up to 5 faulty nodes without becoming disconnected. The conditional node connectivity in this case is therefore 6. We have also shown that the conditional fault-diameter of the *k*-torus is $k + 2$. With this result the torus joins a group of interconnection networks (including the hypercube and the star-graph) whose conditional fault-diameter has been shown to be only two units over the fault-free diameter. We have also illustrated the usefulness of our results by describing two possible fault tolerant routing algorithms for the torus under the forbidden faulty set condition.

6. References

CASADO, R., BERMUDEZ, A., DUATO, J., QUILES, F.J., and SANCHEZ, J.L. 2001. A Protocol for Deadlock-Free Dynamic Reconfiguration in High Speed Local Area Networks, *IEEE Transactions on Parallel and Distributed Systems*, **12(2):** 115-132.

CHEN, C.L., and CHIU, G.M. 2001. A Fault-Tolerant Routing Scheme for Meshes with Non Convex Faults," *IEEE Transactions on Parallel and Distributed Systems*, **12(5):** 467-475.

DAY, K., and AL-AYYOUB, A.E. 1997. Fault Diameter of k-ary n-cube Networks, *IEEE Trans. on Parallel and Distributed Systems*, **8(9):** 903-907.

DAY, K., and AL-AYYOUB, A.E. 2001. Adaptive Fault-Tolerant Routing in Star Networks, *Journal of Interconnection Networks*, **2(2):** 213-231.

ESFAHANIAN, A.H. 1989. Generalized Measures of Fault Tolerance with Application to n-Cube Networks, *IEEE Transactions on Computers,* **38(11):** 1586-1591.

GOMEZ, M., DUATO, J., FLICH, J., LOPEZ, P., ROBLES, A., NORDBOTTEN, N., LYSNE, O., and SKEIE, T. 2004. An Efficient Fault-Tolerant Routing Methodology for Meshes and Tori, *Computer Architecture Letters*, 3, May.

HO, C.T., and STOCKMEYER, L. 2002. A New Approach to Fault-Tolerant Wormhole Routing for Mesh-Connected Parallel Computers, *Proceedings of 16th International Parallel and Distributed Processing Symposium*, April.

IBM BG/L TEAM, An Overview of BlueGene/L Supercomputer, *ACM Supercomputing Conference*, 2002.

LATIFI, S. 1993. Combinatorial Analysis of Fault-Diameter of the n-Cube, *IEEE Transactions on Computers,* **42(1):** 27-33.

LATIFI, S., HEGDE, M., and NARAGHI-POUR, M. 1994. Conditional Connectivity Measures for large Multiprocessor Systems, *IEEE Trans. on Computers,* **43(2):** 218-222.

LYSNE, O., PINKSTON, T., and DUATO, J. 2003. A Methodology for Developing Dynamic Network Reconfiguration Processes, *Proceedings of the International Conference on Parallel Processing*, pp. 77-86.

NORDBOTTEN, N., GOMEZ, M., FLICH, J., LOPEZ, P., ROBLES, A., SKEIE, T., LYSNE, O., and DUSTO, J. 2004. A Fully Adaptive Fault-Tolerant Routing Methodology Based on Intermediate Nodes, *Proceedings of the IFIP International Conference on Network and Parallel Computing*, pp. 341-356.

PINKSTON, T., PANG, R., and DUATO, J. 2003. Deadlock-Free Dynamic Reconfiguration Schemes for Increased Network Dependability, *IEEE Transactions on Parallel and Distributed Systems*, **Vol. 14(8):** 780-794.

ROUSKOV, Y., LATIFI, S., and SRIMANI, P.K. 1996. Conditional Fault Diameter of Star Graph Networks, *Journal of Parallel and Distributed Computing*, **33:** 91-97.

SUH, Y., DAO, B., DUATO, J., and YALAMANCHILI, S. 2000. Software-Based Rerouting for Fault-Tolerant Pipelined Communication, *IEEE Transactions on Parallel and Distributed Systems*, **11(3):** 193-211.

WU, J. 1998. Fault Tolerance Measures for m-Ary n-Dimensional Hypercube Based on Forbidden Faulty Sets, *IEEE Transactions on Computers,* **47(8):** 888-983.

WU, J. 2003. Fault-Tolerant and Deadlock-Free Routing in 2D Meshes Based on Odd-Even Turn Model, *IEEE Transactions on Computers*, **52**, No. 9, September.