On the Routing of the OTIS-Cube Network in Presence of Faults

Ahmad Awwad[1] and Jehad Al-Sadi[2]
[1]Faculty of Computing and Information Technology, Arab Open University, Kuwait
[2]Computer Science Department, Zarqa Private University, Jordan

Abstract *This paper proposes a new fault-tolerant routing algorithm for the well-known class of networks, OTIS-cube. In this new proposed algorithm, each node A starts by computing the first level unsafety set,* S_1^A, *composed of the set of unreachable direct neighbours. It then performs m-1 exchanges with its neighbours to determine the k-level unsafety sets* S_k^A, *for all* $1 \leq k \leq m$, *where m is an adjustable parameter between 1 and 2n + 1. The k-level unsafety set at node A represents the set of all faulty nodes at Hamming distance k from A, which either faulty or unreachable from A due to faulty nodes or links. Equipped with these unsafety sets, we show how each node calculates numeric unsafety vectors and uses them to achieve efficient fault-tolerant routing.*

Keywords: *Interconnection networks, OTIS-cube, fault-tolerant routing algorithm, safety vectors.*

Received July 30, 2003; accepted March 20, 2004

1. Introduction

Recently, there has been an increasing interest in a class of interconnection networks called Optical Transpose Interconnection Systems (OTIS-networks) [3, 24, 28, 30]. The importance of studying the OTIS-Networks stems from the fact that it allows us to define unlimited number of new networks and further study and analyse more deeply some known networks such as star [1], hypercube [22], mesh [19], and arrangement networks [7].

Marsden *et al* were the first to propose the OTIS-Networks [16]. Extensive modelling results for the OTIS have been reported in [11]. The achievable terabit throughput at a reasonable cost makes the OTIS a strong competitor to the electronic alternatives [12, 16]. These encouraging findings prompt the need for further testing of the suitability of the OTIS for real-world parallel applications. A number of computer architectures have been proposed in which the OTIS was used to connect different processors [16]. Krishnamoorthy *et al* [12] have shown that the power consumption is minimised and the bandwidth rate is maximised when the OTIS computer is partitioned into *N* groups of *N* processors each. Due to this fact and its attractive topological properties we will limit our study to this type of OTIS networks.

Furthermore, the advantage of using the OTIS as optoelectronic architecture lies in its ability to manoeuvre the fact that free space optical communication is superior in terms of speed and power consumption when the connection distance is more than few millimetres [12]. In the OTIS, shorter (intra-chip) communication is realised by electronic interconnects while longer (inter-chip) communication is realised by free space interconnects. OTIS technology processors are partitioned into groups, where each group is realised on a separate chip with electronic inter-processor connects. Processors on separate chips are interconnected through free space interconnects. The philosophy behind this separation is to utilise the benefits of both the optical and electronic technologies. Throughout this paper the terms OTIS-computer and OTIS-network will refer to parallel architectures based on the OTIS technology.

Processors within a group are connected by a certain interconnecting topology, while transposing group and processor indexes achieve inter-group links. Figure 1 show a 16 processor OTIS connection where the bold arrows represent an optical links between two processors of two different groups. Using cube as a factor network will yield the OTIS-cube in denoting this network.

OTIS-cube is basically constructed by "multiplying" a cube topology by itself. The set of vertices is equal to the Cartesian product on the set of vertices in the factor cube network. The set of edges *E* in the OTIS-cube consists of two subsets, one is from the factor cube, called *cube*-type edges, and the other subset contains the *transpose* edges. The OTIS approach suggests implementing *cube*-type edges by electronic links since they involve intra-chip short links and implementing transpose edges by free space optics. Throughout this paper the terms "*electronic move*" and the "*OTIS move*" or "*optical move*" will be used to refer to data transmission based on electronic and optical technologies, respectively.

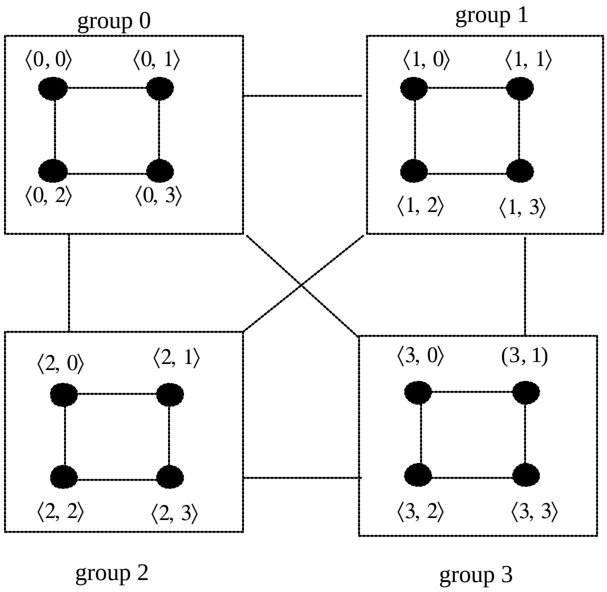


Figure 1. 16-processor OTIS-network.

This paper proposes a fault-tolerant routing algorithm based on the set of unsafety vectors for the "OTIS-cube" network. Most of the existing work on OTIS-cube network focused on its topological properties [6].

2. Related Work

The binary n-cube has been one of the most popular network topologies for multicomputers due to its attractive topological properties, e.g. regular structure, low diameter, and ability to exploit communication locality. Several experimental and commercial systems have been built using the factor cube network including the NCUBE-2 [17], Intel iPSC [20], Cosmic Cube [25], and SGI Origin 2000 multiprocessor [26].

The efficient inter-processor communication is the key to good system performance. The routing algorithm has great impact on network performance, as it is responsible for selecting a network path between two nodes involved in a one-to-one communication. Routing in fault-tolerant and fault-free n-cube (or the cube for short) and its variants has been extensively studied in the past (e. g. see [2, 8, 10, 18, 21, 27]) and hardly you may find any fault-free or even fault-tolerant routing algorithm in OTIS-cube. As the network size scales up the probability of processor and link failure also increases. It is therefore essential to design fault-tolerant routing algorithms that allow to route messages between non-faulty nodes in the presence of faulty components (links and nodes). Few fault-free routing strategies have been proposed in the literature for the cube [4, 5, 9, 13, 14, 15]. Most of these algorithms have assumed that a node knows either only the status of its neighbours (such a model is called *local-information-based*) or the status of all the nodes (*global-information-based*). Local-information-based routing yields sub-optimal routes (if not routing failure) due to the insufficient information upon which the routing decisions are made. Global-information-based routing can achieve optimal or near optimal routing. However, high communication overhead is involved in such algorithms to maintain up-to-date fault information at all network nodes.

The main challenge is to develop a simple and effective way of representing limited global fault information that allows optimal or near-optimal routing. This is the first attempt to design a limited-global-information-based algorithm for the OTIS-cube based on the set of unsafety vectors.

The new proposed limited-global-information-based routing algorithm for the OTIS-cube based on the set of unsafety vectors utilizing the attractive topological properties of OTIS-cube network [6, 23] to achieve an efficient fault-tolerant routing. Each node in OTIS-cube A starts by determining the set of unreachable immediate neighbours due to faulty nodes and links. This set is referred to as the first-level unsafety set at node A and is denoted S_1^A. Then, each node A performs an m-1 exchanges with its immediate neighbours to determine the k-level unsafety set S_k^A for all $1 \leq k \leq m$, where m is an adjustable parameter between 1 and $2n+1$ for an n dimensional OTIS-cube where $2n+1$ is the longest path between any 2 nodes. The k-level unsafety set S_k^A represents the set of all nodes at distance k from A which are faulty or unreachable from node A due to faulty links which causing a network partitioning. Equipped with these unsafety sets, each node calculates numeric unsafety vectors and uses them to achieve efficient fault-tolerant routing algorithm. The larger the value of m is the better the routing decisions are, but at the expense of more computation and communication overhead.

3. Notations and Definitions

The n-dimensional undirected graph binary n-cube Q_n is one of the well known networks which have been used in real life systems [17, 20, 25, 26]. The undirected graph n-cube with 2^n vertices, representing nodes, which are labelled by the 2^n binary strings of length n. Two nodes are joined by an edge if, and only if, their labels differ in exactly one bit position. The label of node A is written $a_n\, a_{n-1} \ldots a_1$, where $a_i \in \{0, 1\}$ is the i-th bit (or bit at i-th dimension) [22].

From the above definition the neighbour of a node A along the i-th dimension is denoted $A^{(i)}$. A faulty n-cube contains faulty nodes and/ or links.

The OTIS-cube is obtained by "multiplying" a cube topology by itself. The vertex set is equal to the *Cartesian* product on the vertex set in the factor cube network. The edge set consists of edges from the factor network and new edges called the *transpose* edges. The formal definition of the OTIS-cube is given below.

Definition 1: Let *cube* = (V_0, E_0) be an undirected graph representing a cube network. The OTIS-*cube* = (V, E) network is represented by an undirected graph obtained from *cube* as follows $V = \{\langle x, y\rangle \mid x, y \in V_0\}$ and $E = \{(\langle x, y\rangle, \langle x, z\rangle) \mid \text{if } (y, z) \in E_0\} \cup \{(\langle x, y\rangle, \langle y, x\rangle) \mid x, y \in V_0\}$ [6].

In the OTIS-cube the address of a node $u = \langle x, y\rangle$ from V is composed of two components. Figure 2 shows a 16 processor OTIS-cube, the notation $\langle g, p\rangle$ is used to refer to the group and processor addresses, respectively. Two nodes $\langle g_1, p_1\rangle$ and $\langle g_2, p_2\rangle$ are connected if, and only if, $g_1 = g_2$ and $(p_1, p_2) \in E_0$ (such that E_0 is the set of edges in cube network) or $g_1 = p_2$ and $p_1 = g_2$, in this case the two nodes are connected by transpose edge.

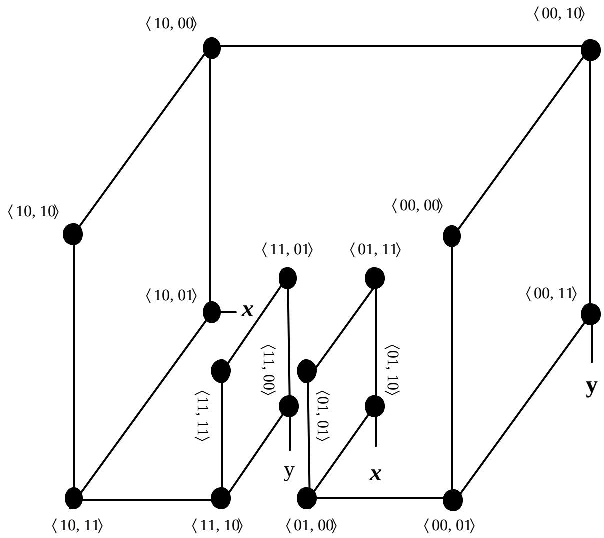


Figure 2. 16-processor OTIS-cube.

The distance in the OTIS-cube is defined as the shortest path between any two processors, $\langle g_1, p_1\rangle$ and $\langle g_2, p_2\rangle$, and involves one of the following forms [29]:

1. When $g_1 = g_2$ then the path involves only electronic moves from source node to destination node.
2. When $g_1 \neq g_2$ and if the number of optical moves is an even number of moves and more than two, then the paths can be compressed into a shorter path of the form: $\langle g_1, p_1\rangle \xrightarrow{E} \langle g_1, p_2\rangle \xrightarrow{O} \langle p_2, g_1\rangle \xrightarrow{E} \langle p_2, g_2\rangle \xrightarrow{O} \langle g_2, p_2\rangle$ where the symbols O and E stand for optical and electronic moves respectively.
3. When g1 ≠ g2, and the path involves an odd number of OTIS moves. In this case the paths can be compressed into a shorter path of the form: $\langle g1, p1\rangle \xrightarrow{E} \langle g1, g2\rangle \xrightarrow{O} \langle g2, g1\rangle \xrightarrow{E} \langle g2, p2\rangle$.

The most important topological properties of the OTIS-cube including the following [6]:

1. *Size*: If the cube factor network of size N, then the size of the OTIS-cube is N^2.
2. *Degree*: Let $\langle g, p\rangle$ be any node in OTIS-cube. Then the degree (or *deg*) of the OTIS-cube is as follows:
$$\deg_{OTIS-CUBE}(g, p) = \begin{cases} \deg_{G_0}(p) & \text{if } g = p \\ \deg_{G_0}(p) + 1 & \text{if } g \neq p \end{cases}$$
3. *Number of Links:* Let N_0 be the number of links and M be the number of nodes in the cube network, then the number of links in the OTIS-*cube* = (M2 - M) / 2 + N_0 * M.
4. *Length*: Let $\langle g_1, p_1\rangle$ and $\langle g_2, p_2\rangle$ be two different nodes in the OTIS-*cube*. To transmit data originated in the source node $\langle g_1, p_1\rangle$ to the destination node $\langle g_2, p_2\rangle$ we follow one of the three possible paths shown above 1, 2, and 3. The length of the shortest path between the nodes $\langle g_1, p_1\rangle$ and $\langle g_2, p_2\rangle$ is:
$$Length = \begin{cases} d(p_1, p_2) & \text{if } g_1 = g_2 \\ min(d(p_1, g_2) + d(p_2, g_1) + 1 & \\ d(p_1, p_2) + d(g_1, g_2) + 2) & \text{if } g_1 \neq g_2 \end{cases}$$
where $d(p_1, p_2)$ is the length of the shortest path between any two processors $\langle g_1, p_1\rangle$ and $\langle g_1, p_2\rangle$.
5. *Diameter*: Let n is the diameter of the cube network, the diameter of the OTIS-cube is $2n + 1$.

4. The Unsafety Vectors Fault-Tolerant Routing Algorithm

In this section, we introduce the adapted fault-tolerant routing algorithm, based on the concept of unsafety sets (defined below). Before presenting the new algorithm, we first discuss how a node in the OTIS-cube calculates its unsafety sets.

The calculation of the unsafety sets is as follows:

Definition 2: The number of direct neighbours *np* of a node *A*, <g_A, p_A>, is defined as:

$$np = \begin{cases} n & \text{if } g_A = p_A \\ n+1 & \text{otherwise} \end{cases}$$

Definition 3: The *first-level unsafety set* S_1^A of a node *A* is defined as

$$S_1^A = \bigcup_{1 \le i \le np} f_A^i, \text{ where } f_A^i \text{ is}$$

given by

$$f_A^i = \begin{cases} \{A^{(i)}\} & \text{if } A^{(i)} \text{ is faulty} \\ f & \text{otherwise} \end{cases}$$

It should be clear that an *isolated node A* is associated with first-level unsafety set containing *np* addresses of faulty nodes, i. e., $|S_1^A| = np$. If for some node *A*, $|S_1^A| = np-1$ then node *A* is called a *dead-end node*.

Each node uses the unsafety set to determine the faulty set F_A, which comprises those nodes which are either faulty or unreachable from A due to faulty nodes or links. This is achieved by performing m-1 exchanges with the reachable neighbours. After determining F_A, node A calculates m unsafety sets denoted S_1^A, $S_2^A, \ldots,$ S_m^A (defined below), where m is an adjustable parameter between 1 and $2n+1$.

Definition 4: The k-level unsafety set $S_k^A, 1 \leq k \leq m$, for node A is given by

$$S_k^A = \{B \in F_A \mid d(A,B) = k\}$$

The k-level unsafety set S_k^A represents node A's view of the set of nodes at distance k from A which are faulty or unreachable from A due to faulty nodes and links. Notice that if the network is disconnected due to faulty nodes and links, A's view about unreachable nodes may not be accurate. In this case massage of Unreachability may occur. Figure 3 gives an outline of the *Find_Unsafety_Sets* algorithm that node A uses it to determine it's faulty and unsafety sets.

Algorithm Find_Unsafety_Sets (<g_A, p_A>: node)
/ called by node A to determine its faulty set F_A */*

S_1^A *= set of faulty or unreachable immediate neighbours;*
$F_A = S_1^A$;
for k := 2 to m do
{
for i :=1 to n do
if $P_A^{(i)} \notin F_A$ *then*
{
send F_A *to* $P_A^{(i)}$;
receive $F_A^{(i)}$ *from* $P_A^{(i)}$;
$F_A = F_A \cup F_A^{(i)}$;
}
if $p_A \neq g_A$
{
send F_A *to* $<g_A, p_A>$;
receive $F_{<g_A, p_A>}$ *from* $<g_A, p_A>$;
$F_A = F_A \cup F_{<g_A, p_A>}$;
}
}
for k := 1 to m do
$S_k^A = \{<g_B, p_B> \in F_A \mid dist(<g_A, p_A>, <g_B, p_B>) = k\}$
End

Figure 3. The find_unsafety_sets algorithm that determines the faulty set for node A.

With respect to a given destination node, D, in a cube network a neighbour $A^{(i)}$ of node A is called a *preferred neighbour* for the routing from A to D if the i-th bit of $A \oplus D$ is 1. We say in this case that i is a *preferred dimension*. Neighbours other than preferred neighbours are called *spare neighbours*. Routing through a spare neighbour increases the routing distance by two over the minimum distance. In general, a preferred neighbour is one step closed to the destination while a spare neighbour increases the routing distance two or more steps over the minimum distance depending of the type of the next move (electronic or optical). An optimal path can be obtained by routing through all preferred dimensions in some order. A node T is called an (A, D)-*preferred transit node* if any preferred dimension for the routing from A to T is also a preferred dimension for the routing from A to D.

Example 1: Consider a two-dimensional OTIS-cube with four faulty nodes (faulty nodes are represented as black nodes), as shown in Figure 4. Table 1 shows the corresponding first-level unsafety set, S_1^A, associated with each node A. The Find_Unsafety_Sets algorithm calculates the sets S_k^A for all $1 \leq k \leq m$ after calculating F_A. To achieve this, (m-1) exchanges of fault information are performed among neighbouring nodes.

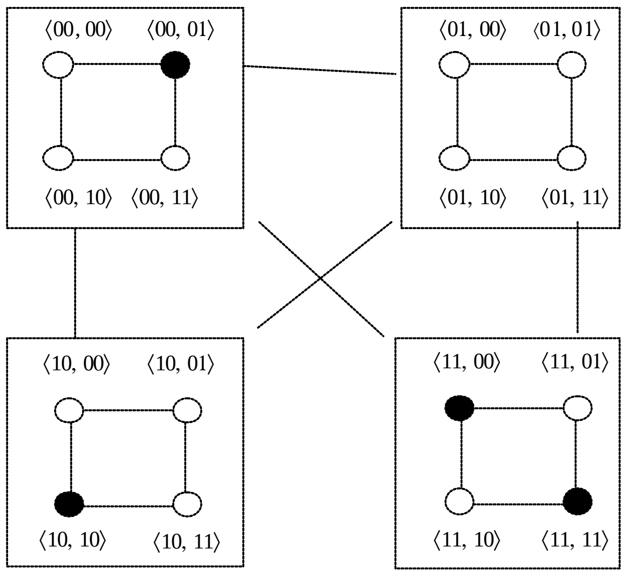


Figure 4. A 2-dimensional OTIS-cube with four faulty nodes.

Table 1. The unsafety sets of nodes in OTIS-cube (n = 2) with 4 faulty nodes.

Node	0000	0001	0010	0011	0100	0101	0110	0111
S_1^A	{1}	Faulty	{}	{1}	{1}	{ }	{ }	{ }

Node	1000	1001	1010	1011	1100	1101	1110	1111
S_1^A	{10}	{ }	Faulty	{10}	Faulty	{12,15}	{12,15}	Faulty

Let $m = 2n + 1$ and for the sake of specific illustrations let us compute the unsafety sets associated with node A = 0000. First, the node assigns the addresses of its immediate faulty neighbours to its faulty set F_A. Then each node performs $2n$ exchanges of the new elements of its faulty set F_A with the immediate non-faulty neighbours. After determining F_A, node A calculates m unsafety sets denoted, S_1^A, S_2^A, …, S_m^A according to the distance between node A and each element of F_A. So, the faulty set for node A in our example, given in decimal representation, $F_A = \{1, 10, 12, 15\}$, and the unsafety sets are $S_1^A = \{1\}$, $S_2^A = \{\}$, $S_3^A = \{10, 12\}$, $S_4^A = \{\}$, and $S_5^A = \{15\}$.

5. The Unsafety Vectors Routing Algorithm

For a given source-destination pair of nodes ($<g_A, p_A>$, $<g_D, p_D>$), we define the *(A, D)-unsafety vector* $U_k^{A,D} = (u_1^{A,D}, \ldots, u_k^{A,D}, \ldots, u_m^{A,D})$ where its k^{th} element is given by $u_k^{A,D} = |\{ T \in S_k^A$, such that T is an (A, D)-preferred transit node$\}|$.

In other words, $u_k^{A,D}$ is the number of faulty or unreachable (A, D)-preferred transit nodes at distance k from $<g_A, p_A>$. $u_k^{A,D}$ can be viewed as a measure of routing unsafety at distance k from $<g_A, p_A>$, hence the name *unsafety vectors* for $U^{A,D}$. We also define an ordering relation '<' for numeric vectors as follows.

Definition 5: For any two numeric vectors $U = (u_1, u_2, \ldots, u_m)$ and $V = (v_1, v_2, \ldots, v_m)$, $U < V$ iff $\exists\, i, 1 \le i \le m$, such that $u_i < v_i$, and $u_j = v_j$ for all $j < i$.

Figure 5 shows the *Unsafety_Vectors* algorithm that each node in the network applies to route a message towards its destination node $<g_D, p_D>$.

Example 2: Consider the cube depicted in Figure 4 where the source node A = 0100, the destination node D = 1011, and let m = 1. According to the unsafety vectors algorithm, the source node A will route message to a preferred neighbor associated with the least number of preferred faulty nodes in its unsafety sets, which is node 0110. By performing the same operations the message will be routed through an optical move to node 1001 then finally to its destination 1011.

Theorem 1: Let $A^{(i)}$ and $A^{(j)}$ be two non faulty (A, D)-preferred neighbours of A. If all preferred neighbours of $A^{(j)}$ are faulty and at least one *preferred* neighbour of $A^{(i)}$ is non faulty then the unsafety vectors algorithm does not route messages of destination D via $A^{(j)}$.

Proof: Since $u_1^{A^{(i)},D} < u_1^{A^{(j)},D}$ then $U^{A^{(i)},D} < U^{A^{(j)},D}$. Therefore, $U^{A^{(j)},D}$ is not the minimal such vector (for the preferred neighbours). □

```
Algorithm Unsafety_Vectors (M: message; <gc, pc>, <gd, pd>: node)
/* called by current node <gc, pc> to route the
   message M to its destination node <gd, pd> */
if  <gc, pc> is source node then
    M.Route_distance = 0
if  Route_distance <= dist(pc, pd)+dist(gc, gd)+
 (2n+1 )* No_FaultyNodes  then
   {
     M.Route_distance:= M.Route_distance+1
     if  (gc = gd )and (pc = pd) then
          exit; /* destination reached */
     if  gc = gd  then
          route(<gc, pc>,<gd, pd>)  /* curr& dest.at the
          same group */
     if  (dist(pc, pd)+dist(gc, gd)+2)< dist(pc,gd)+dist(gc,
     pd)+1) and the two optical moves (gcpd?  pdgc,
     pdgd?  gdpd) are not faulty then
        {
          if  pc= pd  then
             move m to < pc, gc>
          else  route(<gc, pc>,< gc, pd>)
        }
     else if the optical move (gcgd?  gdgc) is not faulty
     then
         {
           if  pc = gd  then
              move m to < pc, gd>
           else  route(<gc, pc>,<gc, gd>)
         }
        else if  gc  ? pc and the node<pc, gc > is not
          faulty  then
             send M to <pc, gc >    /* disturb the
             message*/
        else  looping
   }
End.

Function route(<gc, pc>,<gd, pd>:node)
{
if $  a preferred non-faulty neighbour  A(i) with least
 (A(i), D)-unsafety vector U^A(i),D  and  A(i)  is not ad-end
 then
     send M to  A(i)
else if $  a spare non-faulty neighbour A(j) with
  least (A(j), D)-unsafety vector U^A(j),D  and A(j) is
  not dead-end then
     send M to  A(j)
else if  gc ? pc and the node<pc, gc > is not faulty then
     send M to <pc, gc >    /* disturb the message*/
else failure        /* destination unreachable */
}
```

Figure 5. A description of the proposed unsafety vectors routing algorithm.

6. Conclusion

This paper has proposed a new fault-tolerant routing based on the concept of unsafety vectors. As a first step in this algorithm, each node A determines its view of the faulty set F_A of nodes, which are either faulty or unreachable from A. This is achieved by performing at most $2n$ exchanges with the reachable neighbours. After determining F_A, node A calculates m unsafety sets denoted , $S_1^A, S_2^A, \ldots, S_m^A$ where m is an adjustable parameter between 1 and $2n+1$. The m-level unsafety set represents the set of all nodes at distance m from A which are faulty or unreachable from A due to faulty links or nodes.

Equipped with these unsafety sets each node calculates unsafety vectors and uses them to achieve fault-tolerant routing in the OTIS-cube. The larger the value of m is the better the routing decisions are, but at the expense of more communication overhead. An extension for this work is to implement the proposed routing algorithm for all the different network sizes and conduct a performance analysis through extensive simulation experiments to show the superiority of the proposed algorithm using the set of unsafety vectors.

References

[1] Akers S. B., Harel D., and Krishnamurthy B., "The Star Graph: An Attractive Alternative to the N-Cube," *in Proceedings of the International Conference Parallel Processing*, pp. 393-400, 1987.

[2] Al-Sadi J., Day K., and Ould-Khaoua M., "Unsafety Vectors: A New Fault-Tolerant Routing for the Binary N-Cube," *Journal of Systems Architecture*, vol. 47, no. 9, pp. 783-793, 2002,

[3] Awwad A., Al-Ayyoub A., and Ould-Khaoua M., "Efficient Routing Algorithms on the OTIS–Networks," *in Proceedings of the 3rd International Conference on Information Technology* (ACIT'2002), The University of Qatar, Doha, pp. 138-144, December 2002.

[4] Chen M. S. and Shin K. G., "Adaptive Fault-Tolerant Routing in Hypercube Multicomputers," *IEEE Transactions Computers*, vol. 39, no.12, pp.1406-1416, 1990.

[5] Chen M. S. and Shin K. G., "Depth-First Search Approach for Fault-Tolerant Routing in Hypercube Multicomputers," *IEEE Transactions Parallel and Distributed Systems*, vol. 1 , no. 2, pp. 152-159, 1990.

[6] Day K. and Al-Ayyoub A., "Topological Properties of OTIS-Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 4, pp. 359-366, 2002.

[7] Day K. and Tripathi A., "Arrangement Graphs: A Class of Generalised Star Graphs," *Information Processing Letters*, vol. 42, pp. 235-241, 1992.

[8] Gaughan P. T., Yalamanchili S., "Adaptive Routing Protocols for Hypercube Interconnection Networks," *Computer*, vol. 26, no. 5, pp. 12-24, 1993.

[9] Gordon J. M. and Stout Q. F., "Hypercube Message Routing in the Presence of Faults," *in Proceedings of the 3rd Conference Hypercube Concurrent Computers and Applications*, pp. 251-263, 1988.

[10] Graham S. and Seidel S., "The Cost of Broadcasting on Star Graphs and K-Ary Hypercubes," *IEEE Transactions Computers*, vol. 6, pp. 756-759, 1993.

[11] Hendrick W., Kibar O., Marchand P., Fan C., Blerkom D., McCormick F., Cokgor I., Hansen M., and Esener S., "Modeling and Optimisation of the Optical Transpose Interconnection System," *in Optoelectronic Technology Centre*, Cornell University, September 1995.

[12] Krishnamoorthy A., Marchand P., Kiamilev F., and Esener S., "Grain-size Considerations for Optoelectronic Multistage Interconnection Networks," *Applied Optics*, vol. 31, no. 26, pp. 5480- 5507, 1992.

[13] Lan Y., "A Fault-Tolerant Routing Algorithm in Hypercubes," *in Proceedings of the International Conference Parallel Processing*, pp. 163-166, 1994.

[14] Lee T. C. and Hayes J. P., "A Fault-Tolerant Communication Scheme for Hypercube Computers," *IEEE Transactions Computers*, vol. 41, no. 10, pp. 1242-1256, 1992.

[15] Linder D. and Harden J., "An Adaptive and Fault Tolerant Wormhole Routing Strategy for K-Ary Hypercubes," *IEEE Transactions Computers*, vol. 1, pp. 2-12, 1991.

[16] Marsden G., Marchand P., Harvey P., and Esener S., "Optical Transpose Interconnection System Architecture," *Optics Letters*, vol.18, no.13, pp. 1083-1085, 1993.

[17] N-Cube Systems, *N-cube Handbook*, 1986.

[18] Ni M. and McKinley P. K., "A Survey of Routing Techniques in Wormhole Networks," *Computer*, vol. 26, no.2, pp. 62-76, 1993.

[19] Ranka S., Wang J., and Yeh N., "Embedding Meshes on the Star Graph," *in Proceedings of the Supercomputing*'90," New York, USA, pp. 476-485, 1990.

[20] Rattler J., "Concurrent Processing: A new Direction in Scientific Computing," *in Proceedings of the AFIPS Conference*, pp. 157-166, 1985.

[21] Saad Y. and Schultz M. H., "Data Communication in Hypercubes," *Technical Report* YALEU/ DCS/ RR-428, Yale University, 1985.

[22] Saad Y. and Schultz M. H., "Topological Properties of Hypercubes," *IEEE Transactions Computers*, vol. 37, no. 7, pp. 867-872, 1988.

[23] Sahni S. and Wang C., "BPC Permutations on the OTIS-hypercube Optoelectronic Computer," *Informatica*, vol. 22, pp. 263-269, 1998.

[24] Sahni S. and Wang C., "BPC Permutations on the OTIS-mesh Optoelectronic Computer," *Technical Report*, University of Florida, 1997.

[25] Seitz C. L., "The cosmic cube," *Communications of ACM*, vol. 28, no.1, pp. 22-23, Jan 1985.

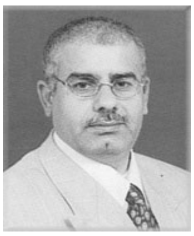
[26] Silicon Graphics, "Origin 200 and Origin 2000," *Technical Report*, 1996.

[27] Sullivan H., Bashkow T., and Klappholz D., "A Large Scale, Homogeneous, Fully Distributed Parallel Machine," *in Proceedings of the 4th Annual Symposium Computer Architecture*, pp. 105-124, 1977.

[28] Wang C. and Sahni S., "Basic Operations on the OTIS-mesh Optoelectronic Computer," *IEEE Transactions Parallel and Distributed Systems*, vol. 9, no. 12, pp.1226-1236, 1998.

[29] Wang C. F., "Algorithms for the OTIS Optoelectronic Computers," *PhD Thesis*, University of Florida, 1998.

[30] Zane F., Marchand P., Paturi R., and Esener S., "Scalable Network Architecture Using the Optical Transpose Interconnection System (OTIS)," *Journal of Parallel and Distributed Computing*, vol. 60, pp. 521-538, 2000.



Ahmad Awwad finished both of his BSc degree in computer science and his MSc degree in mathematical science from Tennessee State University, USA, in 1987 and 1989, respectively. He gained his PhD in computer science from the University of Glasgow, United Kingdom, 2002. He served in Oman as senior lecturer at the Sohar College, and at Zarqa Private University as Head of Computer Science Department. He is currently a staff-tutor in the Faculty of Information Technology at Arab Open University. His research interests are parallel processing, interconnection networks, product networks, opto-electronic networks, algorithms design for HSPC, and distance learning. He is a member of the IEEE-CS.

Jehad Al-Sadi received his BSc degree in computer science from Tennessee State University, USA, in 1989, and MSc degree in Computer Science from Jackson State University, USA, in 1993. From graduation till 1996, he was a lecturer in the Department of Computing Science at Al-Isra University, Jordan. He gained his PhD in computer science from the University of Glasgow, U.K, 2002. He is currently the chairman of the Computer Science Department at Zarqa Private University, Jordan. His research interests are multicomputers, interconnection networks, optical networks, and fault-tolerant routing. He is a member of IEEE.