

Symmetric Tori Connected Torus Network

Faiz Al Faisal and M.M. Hafizur Rahman

Dept. of CSE, KUET, Khulna- 9203, Bangladesh
dipu_7009@yahoo.com, hafiz90305@gmail.com

Abstract

A Symmetric Tori connected Torus Network (STTN) is a 2D-torus network of multiple basic modules, in which the basic modules are 2D-torus networks that are hierarchically interconnected for higher-level networks. In this paper, we present the architecture of the STTN, addressing of node, routing of message, and evaluate the static network performance of STTN, TTN, TESH, mesh, and torus networks. It is shown that the STTN possesses several attractive features, including constant degree, small diameter, low cost, small average distance, moderate bisection width, and high fault tolerant performance than that of other conventional and hierarchical interconnection networks.

Keywords: STTN, Node Degree, Diameter, Cost, Average Distance, Bisection Width, and Arc Connectivity.

I. INTRODUCTION

Sequential computer steadily increases their speed to meet the computation demand, and it has already been reached saturated. Thus, the only way to meet the increasing demand of computation power to solve the grand challenge problems is to use parallel computers. Massively Parallel Computer (MPC) systems with thousands of nodes have been commercially available and efforts have been made to build MPC systems with millions of nodes. In such computers, with millions of nodes, the large diameter of conventional topologies is completely infeasible. Hierarchical interconnection networks (HIN) [1] are a cost-effective way to interconnect a large number of nodes. A variety of hypercube-based HIN have been proposed [2-5], but for MPC systems, the number of physical links becomes prohibitively large. To alleviate this problem, several k-ary n-cube based HIN have been proposed. However, the performance of these networks does not yield any obvious choice of an interconnection network for MPC. No one is clear winner in all aspect of network design.

A Tori connected mESH (TESH) network [6,7] is an HIN aiming for large-scale 3D MPC systems, consisting of multiple basic modules (BMs) which are 2D-mesh networks. The BMs are hierarchically interconnected by a 2D-torus to build higher level networks. The restricted use of physical links between BMs in the higher level networks and within the BMs reduces the dynamic communication performance of this network [8]. It has already been shown that a torus network has better dynamic communication performance than a mesh network [9]. We have replaced the 2D-mesh of a TESH

network by a 2D-torus network, and the modified HIN is called Tori-connected Torus Network (TTN) [10]. It is seen that TTN is suitable for a few tens of thousands of node [11]. For millions of nodes, TTN does not give better performance. The assignment of free links of BM for higher level interconnection is asymmetric in the TTN. We assign the free links in a symmetric order for higher level interconnection. This new interconnection network is called Symmetric Tori connected Torus Network (STTN). It provides scalability up to a million of nodes with less cost.

In this paper, we address the architectures of the STTN and evaluate its static network performance. The static network performance will be evaluated in terms of node degree, network diameter, cost, average distance, bisection width, and arc connectivity. The main objective is generalized study of static network performance for the STTN.

The remainder of the paper is organized as follows. In Section II, we briefly describe the basic architecture of the STTN. Addressing of nodes of the STTN and the routing of messages are discussed in Section III and Section IV, respectively. The static network performance of the STTN is discussed in Section V. Finally, in Section VI, we conclude this paper.

II. INTERCONNECTION OF THE STTN

The Symmetric Tori connected Torus Network (STTN) is a hierarchical interconnection network consisting of multiple basic modules (BM) that are hierarchically interconnected to form a higher level network. A ($2^m \times 2^m$) BM consists of a 2D-torus network of 2^{2m} processing elements (PE) having 2^m rows and 2^m columns, where m is a positive integer. Considering $m=2$, a BM of size (4×4) is depicted in Fig. 1(a). Each BM has 2^{m+2} free ports at the contours for higher level interconnection. All ports of the interior nodes are used for intra-BM connections. All free ports of the exterior nodes, either one or two, are used for inter-BM connections to form higher level networks. In this paper, BM refers to a Level-1 network.

Successive higher level networks are built by recursively interconnecting (2^{2m}) immediate lower level subnetworks in a ($2^m \times 2^m$) 2D-torus network. As portrayed in Fig. 1(b), considering ($m = 2$) a Level-2 STTN can be formed by interconnecting $2^{2 \times 2} = 16$ BMs. Similarly, a Level-3 network can be formed by interconnecting 16 Level-2 subnetwork and so on. Each

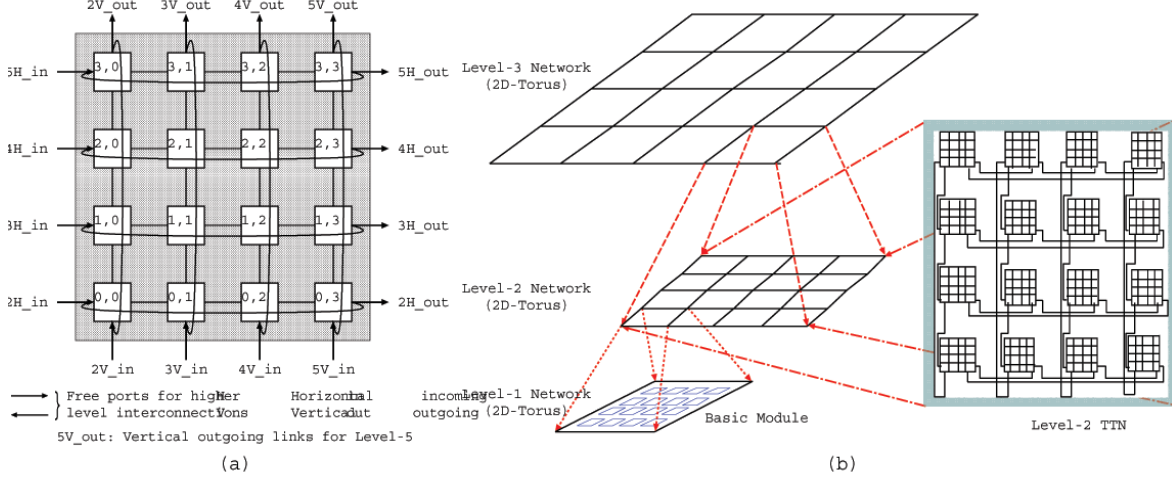


Fig. 1. Interconnection of a STTN (a) Basic module (b) Higher level network

BM is connected to its logically adjacent BMs. To avoid clutter, the wraparound links of the BMs are not shown. It is useful to note that for each higher level interconnection, a BM uses $4 \times (2^q) = 2^{q+2}$ of its free links, $2(2^q)$ free links for vertical interconnections and $2(2^q)$ free links for horizontal interconnections. Here, $q \in \{0, 1, \dots, m\}$, is the inter-level connectivity. $q = 0$ leads to minimal inter-level connectivity, while $q = m$ leads to maximum inter-level connectivity. As shown in Figure 1(a), for example, the (4×4) BM has $2^{2+2} = 16$ free ports. If we chose $q = 0$, then $4(2^0) = 4$ of the free ports and their associated links are used for each higher level interconnection, 2 for horizontal and 2 for vertical interconnection. Among these 2 links, one is used for incoming link and another one for used for outgoing link, i.e., a single links is used for vertical_in, vertical_out, horizontal_in, and horizontal_out.

A STTN (m, L, q) is constructed using $2^m \times 2^m$ BMs, has L levels of hierarchy with inter-level connectivity q . In principle, m could be any positive integer value. However, if $m = 1$, then the network degenerates to a hypercube network. Hypercube is not a suitable network, because its node degree increases along with the increase of network size. If $m = 2$, then it is considered the most interesting case, because it has better granularity than the large BMs. If $m \geq 3$, the granularity of the family of networks is coarse. If $m = 3$, then the size of the BM becomes (8×8) with 64 nodes. Correspondingly, the Level-2 network would have 64 BMs. In this case, the total number of nodes in a Level-2 network is $N = 2^{2 \times 3 \times 2} = 4096$ nodes, and Level-3 network would have 262144 nodes. Clearly, the granularity of the family of networks is rather coarse. In the rest of this paper we consider $m = 2$, therefore, we focus on a class of STTN $(2, L, q)$ networks.

The highest level network which can be built from $(2^m \times 2^m)$ BM is $L_{\max} = 2^{m-q} + 1$. With $q = 0$ and $m = 2$, $L_{\max} = 5$, Level-5 is the highest possible level. The total number of nodes in a STTN having $2^m \times 2^m$ BMs is $N = 2^{2mL}$. Using maximum level of hierarchy, $L_{\max} = 2^{m-q} + 1$, the maximum number of nodes which can be

interconnected by a STTN (m, L, q) is $N = 2^{2m(2m-q+1)}$. For the case of (4×4) BM with $q = 0$, a network consists of 1 million nodes.

The question may arise, whether we need massively parallel computers with thousands of nodes or millions of nodes. The answer is 'yes'. Solving the most challenging problems in many areas of science and engineering, such as defense (maintaining national security), aerospace (space exploration and shuttle operation), disaster management (recovering from natural disaster), and weather forecasting (predicting and tracking severe weather), requires teraflop performance for more than a thousand hours at a time. This is why, in the near future, we will need computer systems capable of computing at the tens of petaflops level or even exaflops level. To achieve this level of performance, we need MPC system with thousands or millions of nodes.

III. ADDRESSING OF NODES

Base-4 numbers are used for convenience of address representation. As seen in Figure 1(a), nodes in the BM are addressed by two digits, the first representing the row index and the next representing the column index. More generally, in a Level- L STTN, the node address is represented by:

$$\begin{aligned}
 A &= A^L A^{L-1} A^{L-2} \dots \dots A^2 A^1 \\
 &= a_{n-1} a_{n-2} a_{n-3} \dots \dots a_2 a_1 a_0 \\
 &= a_{2L-1} a_{2L-2} a_{2L-3} a_{2L-4} \dots \dots a_3 a_2 a_1 a_0 \\
 &= (a_{2L-1} a_{2L-2}) (a_{2L-3} a_{2L-4}) \dots \dots (a_3 a_2) (a_1 a_0) \quad (1)
 \end{aligned}$$

Here, the total number of digits is $n = 2L$, where L is the level number. A^L is the address of level L and left $(a_{2L-1} a_{2L-2})$ is the co-ordinate position of Level- $(L-1)$ for Level- L network. Pairs of digits run from group number 1 for Level-1, i.e., the BM, to group number L for the L -th level. Specifically, l -th group $(a_{2L-1} a_{2L-2})$ indicates the location of a Level- $(l-1)$ subnetwork within the l -th group to which the node belongs; $1 \leq l \leq L$. In a two-level network the address becomes $A = (a_4 a_3) (a_1$

a_0). The first pair of digits ($a_4 a_3$) identifies the BM to which the node belongs, and the last pair of digits ($a_1 a_0$) identifies the node within that BM.

The assignment of inter-level ports for the higher level networks has been done quite carefully so as to minimize the higher level traffic through the BM. The address of a node n^1 encompasses in BM_1 is represented as $n^1 = (a_{2L-1}^1 a_{2L-2}^1 \dots \dots \dots a_3^1 a_2^1 a_1^1 a_0^1)$. The address of a node n^2 encompasses in BM_2 is represented as $n^2 = (a_{2L-1}^2 a_{2L-2}^2 \dots \dots \dots a_3^2 a_2^2 a_1^2 a_0^2)$. The node n^1 in BM_1 and n^2 in BM_2 are connected by a link if the following condition is satisfied.

$$\exists i \{ a_i^1 = (a_i^2 \pm 1) \bmod 2^m \wedge \forall j (j \neq i \rightarrow a_j^1 = a_j^2) \} \\ \text{where } i, j \geq 2 \quad (2)$$

IV. ROUTING ALGORITHM FOR STTN

Routing of messages in the STTN is performed from top to bottom as in TTN [10]. That is, it is first done at the highest level network; then, after the packet reaches its highest level sub-destination, routing continues within the subnetwork to the next lower level sub-destination. This process is repeated until the packet arrives at its final destination. When a packet is generated at a source node, the node checks its destination. If the packet's destination is the current BM, the routing is performed within the BM only. If the packet is addressed to another BM, the source node sends the packet to the outlet node which connects the BM to the level at which the routing is performed.

```

Routing TTN(s,d);
source node address:  $s_{2L-1}, s_{2L-2}, s_{2L-3}, \dots, s_1, s_0$ 
destination node address:  $d_{2L-1}, d_{2L-2}, d_{2L-3}, \dots, d_1, d_0$ 
tag:  $t_{2L-1}, t_{2L-2}, t_{2L-3}, \dots, t_1, t_0$ 
for  $i = 2L - 1 : 2$ 
  if  $\{(d_i - s_i + 2^m) \bmod 2^m\} \leq \frac{2^m}{2}$  then routedir = positive;
     $t_i = \{(d_i - s_i + 2^m) \bmod 2^m\}$ ;
  else routedir = negative;
     $t_i = \{2^m - (d_i - s_i + 2^m) \bmod 2^m\}$ ; endif;
   $g = \text{get\_group\_number}(s, d, \text{routedir})$ ;
  while  $(t_i \neq 0)$  do
    if  $(i \bmod 2) = 0$ , then
       $\text{outlet\_node}_x = \text{outlet}_x(g, \lfloor \frac{i}{2} + 1 \rfloor, H, \text{routedir})$ ;
       $\text{outlet\_node}_y = \text{outlet}_y(g, \lfloor \frac{i}{2} + 1 \rfloor, H, \text{routedir})$ ; endif;
    if  $(i \bmod 2) = 1$ , then
       $\text{outlet\_node}_x = \text{outlet}_x(g, \lfloor \frac{i}{2} + 1 \rfloor, V, \text{routedir})$ ;
       $\text{outlet\_node}_y = \text{outlet}_y(g, \lfloor \frac{i}{2} + 1 \rfloor, V, \text{routedir})$ ; endif;
    BM_Routing( $\text{outlet\_node}_x, \text{outlet\_node}_y$ )
    if (routedir = positive), move packet to next BM; endif;
    if (routedir = negative), move packet to previous BM; endif;
    if  $(t_i > 0)$ ,  $t_i = t_i - 1$ ; endif;
    if  $(t_i < 0)$ ,  $t_i = t_i + 1$ ; endif;
  endwhile;
endfor;
BM_Routing( $t_y, t_x$ )
end
BM_Routing( $t_1, t_0$ );
BM_tag  $t_1, t_0 = \text{receiving node address } (r_1, r_0) - \text{destination } (d_1, d_0)$ 
for  $i = 1 : 0$ 
  if  $(t_i > 0 \text{ and } t_i \leq 2^{m-1})$  or  $(t_i < 0 \text{ and } t_i \geq -(2^{m-1}))$ , movedir = posi;
  if  $(t_i > 0 \text{ and } t_i = (2^m - 1))$  or  $(t_i < 0 \text{ and } t_i \geq -2^{m-1})$ , movedir = neg;
  if (movedir = positive and  $t_i > 0$ ), distance =  $t_i$ ; endif;
  if (movedir = positive and  $t_i < 0$ ), distance =  $2^m + t_i$ ; endif;
  if (movedir = negative and  $t_i < 0$ ), distance =  $t_i$ ; endif;
  if (movedir = negative and  $t_i > 0$ ), distance =  $-2^m + t_i$ ; endif;
endfor
while  $(t_1 \neq 0$  or distance $_1 \neq 0)$  do
  if (movedir = positive), move packet to  $+y$  node; distance $_1 = \text{distance}_1$ ;
  if (movedir = negative), move packet to  $-y$  node; distance $_1 = \text{distance}_1$ ;
endwhile;

```

Fig. 2. Routing algorithm of the STTN

Due to simplicity and fast routing, we have considered the dimension order routing algorithm for the STTN. At each level, vertical routing is performed first. Once the packet reaches the correct row, then horizontal routing is performed. Routing in the STTN is strictly defined by the source node address and the destination node address. Let a source node address be $s = (s_{2L-1}, s_{2L-2}, (s_{2L-3}, s_{2L-4}), \dots, (s_3, s_2), (s_1, s_0))$, a destination node address be $d = (d_{2L-1}, d_{2L-2}, (d_{2L-3}, d_{2L-4}), \dots, (d_3, d_2), (d_1, d_0))$, and a routing tag be $t = (t_{2L-1}, t_{2L-2}), (t_{2L-3}, t_{2L-4}), \dots, (t_1, t_0)$, where $t_i = d_i - s_i$. Figure 2 shows the routing algorithm for the STTN. The function *get_group_number* gets a group number. Arguments of this function are s, d , and routing direction. Each free-link is labeled as (g, l, d, δ) , where $2 \leq l \leq L$ is the level, $d \in \{V, H\}$ is the dimension, and $\delta \in \{+, -\}$ is the direction. The functions *outlet_x* and *outlet_y* results the outlet node of the BM for higher level.

V. STATIC NETWORK PERFORMANCE

Although the actual performance of a network depends on many technological and implementation issues, several topological properties and performance metrics can be used to evaluate and compare different network topologies in a technology-independent manner. Most of these properties are derived from the graph model of the network topology. In this section, we discuss some performance metrics that characterize the cost and performance of an interconnection network. For the performance evaluation, we have considered mesh, torus, TESH network, TTN, and proposed STTN.

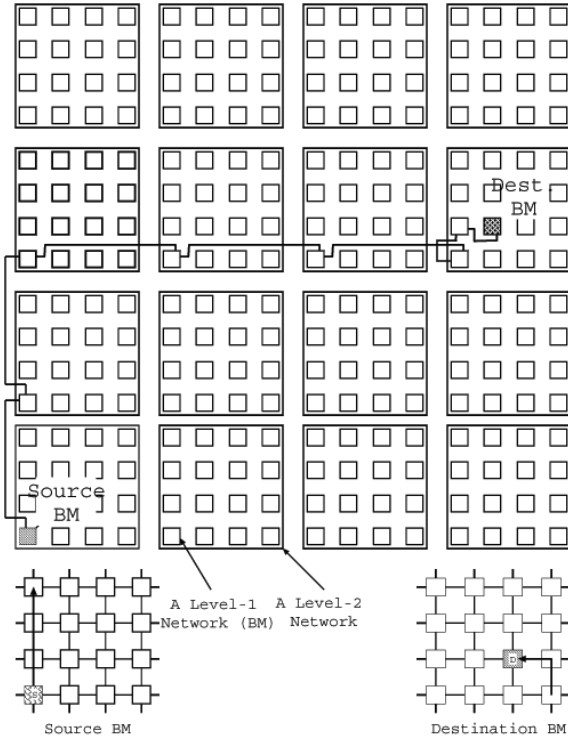


Fig. 3. An example of routing algorithm of the STTN

A. NODE DEGREE

The *node degree* is defined as the maximum number of physical links emanating from a node. Since each exterior node of the BM has six links, the degree of the STTN is 6, and it is independent of network size. Constant degree networks are easy to expand, and the network interface cost of a node remains unchanged with increasing network size. The I/O interface cost of a particular node is proportional to its degree. It is shown in Table 1 that the degree of the STTN exactly equal to that of TTN and is higher than that of mesh, torus, and TESH networks.

B. DIAMETER

The *diameter* of a network is the maximum inter-node distance, i.e., the maximum number of links that must be traversed to send a message to any node along the shortest path. As a definition, the distance between adjacent nodes is unity. Diameter is the maximum distance among all distinct pairs of nodes along the shortest path. The diameter is commonly used to describe and compare the static network performance of the network's topology. Networks with small diameters are preferable. The smaller the diameter of a network, the shorter the time to send a message from one node to the node that is farthest away from it. The diameter of the STTN with $q=0$ is calculated using the following equations:

$$D_{STTN(m,L,0)} = D_{BM(s)} + \max \sum_{i=L}^2 (D_{BM}^{level-move} + D_i) + D_{BM(d)}. \quad (3)$$

$D_{BM(s)} = 4$ is the maximum number of hops from the source to the highest level outgoing node. $D_{BM}^{level-move}$ is the number of hops for the immediate lower level outgoing node. D_i is the number of hops in the Level- i routing. $D_i = 7$ for Level-2 and Level-3 routing. $D_i = 13$ for Level-4 and Level-5 routing, because forward and backward nodes are separated by one node-distance. $D_{BM(d)} = 4$ is the maximum number of hops from the incoming nodes of destination BM to the destination. $D_{BM(s)}$ and $D_{BM(d)}$ are the diameter of a $(2^m \times 2^m)$ torus network.

We have evaluated the diameter of the STTN, TTN, and TESH network by simulation and mesh and torus network by their static formula and the result is plotted in Fig. 4. Clearly, the STTN has a much smaller diameter than TESH, torus, and mesh networks. And it is slightly higher than that of TTN. However, it is shown that the difference is diminishing with the increase of number of nodes.

C. COST

Inter-node distance, message traffic density, and fault-tolerance are dependent on the diameter and the node degree. The product (diameter \times node degree) is a good criterion for measuring the relationship between cost

and performance of a multiprocessor system [3]. An interconnection network with a large diameter has a very low message passing bandwidth, and a network with a high node degree is very expensive. In addition, a network should be easily scalable; there should be no changes in the basic node configuration as we increase the number of nodes. The cost of different networks is plotted in Fig. 5, and it is shown that the cost of STTN is far lower than that of mesh and torus networks, and slightly higher than that of TTN and TESH networks.

D. AVERAGE DISTANCE

The *average distance* is the mean distance between all distinct pairs of nodes in a network. A small average distance results small communication latency, especially for distance-sensitive routing, such as store and forward. But it is also crucial for distance-insensitive routing, such as wormhole routing, since short distances imply the use of fewer links and buffers, and therefore less communication contention. We have evaluated the average distances for STTN, TTN, and TESH network by simulation and mesh and torus networks by their corresponding formulae and the result is plotted in Fig. 6. It is shown that the average distance of the STTN is remarkably lower than that of TESH network, and far lower than that of mesh and torus networks. With large number of nodes, i.e., $2^{20} = 1$ million nodes, the average distance of the STTN is lower than that of its rival TTN. Although the dynamic communication performance of a program on a multicomputer depends on the actual times taken for data transfer, a smaller average distance and diameter of an interconnection network yields a smaller communication latency of that network.

E. BISECTION WIDTH

The *Bisection Width (BW)* of a network is defined as the minimum number of links that must be removed to partition the network into two equal halves. Many problems can be solved in parallel using *binary divide-and-conquer*: split the input data set into two halves, and solve them recursively on both halves of the interconnection network in parallel, then merge the results from both halves into the final result. Small bisection width implies low bandwidth between the two halves, and it can slow down the final merging phase. On the other hand, a large bisection width is undesirable for the VLSI design of the interconnection network, since it implies a lot of *extra chip wires*, such as in hypercube [6]. The bisection width of the STTN(m, L, q) is given by:

$$BW_{STTN(m, L, q)} = 2^m \times 2^{\{2m(L-1)-1\}} \times 2 = 2^{m(2L-3)+1} \quad (4)$$

It is calculated by counting the number of links that need to be removed to partition the highest level (Level- L) torus network. We have calculated the average distance of STTN, TTN, TESH, mesh, and torus networks by their respective static formula and it is here

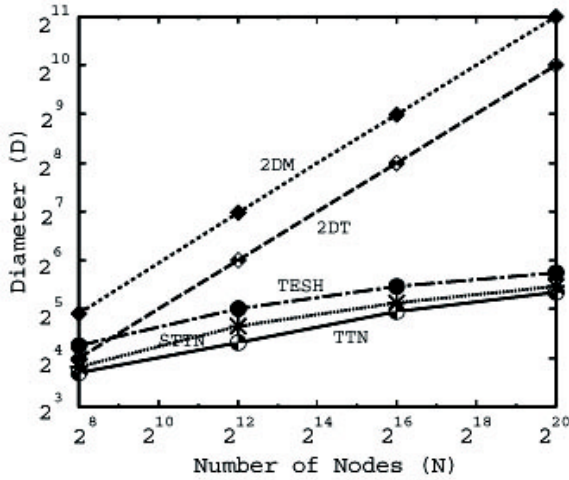


Fig. 4. Diameter of networks as a function of No. of nodes (N).

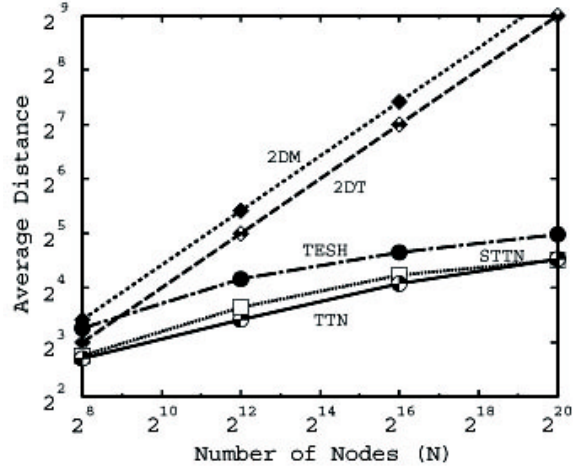


Fig. 6. Average distance of networks as a function of No. of nodes (N).

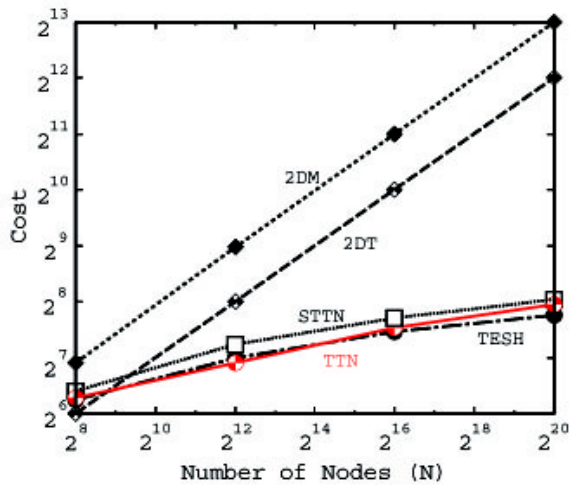


Fig. 5. Cost of networks as a function of No. of nodes (N).

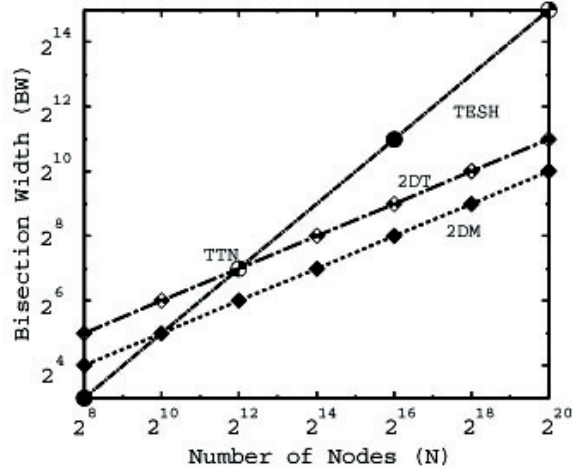


Fig. 7. Bisection Width of networks as a function of No. of nodes (N).

plotted in Fig. 7. It is shown that the bisection width of the STTN is exactly equal to that of the TTN and TESH network. And after 4096 node, it is higher than that of conventional mesh and torus networks.

F. ARC CONNECTIVITY

Arc Connectivity measures the robustness of a network. It is a measure of the multiplicity of paths between processors. Arc connectivity is the minimum number of links that must be removed in order to break the network into two disjoint parts. High arc connectivity improves performance during normal operation by avoiding link congestion, and also improves fault tolerance. The ratio between arc connectivity and the degree of a node gives a measure of static fault tolerance performance. A network is maximally fault-tolerant if its connectivity is equal to the degree of that

network. The arc connectivity of various networks is shown in Table 1. Clearly, the arc connectivity of the STTN is exactly equal to that of TTN and torus network and higher than that of the mesh and TESH networks. However, the arc connectivity of the torus network is exactly equal to its degree. Thus, torus is more fault tolerant than all the networks. STTN is exactly equal fault tolerant to that of TTN and more fault tolerant than mesh and TESH networks.

Table 1. Comparison of degree and arc connectivity for various networks

	2D M	2DT	TESH	TTN	STTN
Degree	4	4	4	6	6
Arc Connectivity	2	4	2	4	4

The operating speed of a network is limited by the

corresponding physical length of links. With 2D-planar implementation, the maximum lengths of Level-2 and Level-3 STTN are 12 and 48, respectively. These are the wrap-around links of the higher level interconnection. The BM of STTN is a 2D-torus network. Thus, we need some more medium length links whose length is 2^m-1 . The main demerit of STTN is that we need some medium and high length links. However, this cost yields better performance. To overcome this problem, we have kept in mind as future work, the replacement of the electronic links by optical links, i.e., to study the architecture and performance of opto-electronic-STTN or hybrid-STTN.

VI. CONCLUSION

A new hierarchical interconnection network, called Symmetric Tori connected Torus Network (STTN), is proposed for the high performance MPC systems. The architecture of the STTN, addressing of nodes, and routing of messages were discussed in detail. We have evaluated the static network performance of the STTN, as well as that of several other interconnection networks. From the static network performance, it has been shown that the STTN possesses several attractive features, including constant node degree, small diameter, low cost, small average distance, better bisection width, and better fault tolerant performance. The diameter and average distance of the STTN is lower than that of TTN, TESH, torus, and mesh networks for very large size network. STTN is equal fault tolerant to that of TTN and more fault tolerant than mesh and TESH networks. The STTN yields better static network performances with reasonable cost for a network consist of millions of nodes, which are indispensable for next generation high-performance MPC. Therefore, STTN would be a good choice of an interconnection network for millions of nodes.

This paper focused on the architectural structure and static network performance. Issues for future work include the following: (1) evaluation of dynamic communication performance using dimension order routing and (2) assessment of the performance improvement of the STTN with an adaptive routing algorithm.

ACKNOWLEDGEMENT

The authors are grateful to the anonymous reviewers for their constructive comments which helped us greatly to improve the clarity of this paper.

REFERENCES

- [1] Y.R. Potlapalli, "Trends in Interconnection Network Topologies: Hierarchical Networks", *Int'l. Conf. on Parallel Processing Workshop*, 1995, pp. 24–29.
- [2] A. El-Amawy and S. Latifi, "Properties and Performance of folded Hypercube", *IEEE Trans on Parallel and Distributed Systems*, Vol. 2, no. 1, 1991, pp. 31–42.
- [3] J.M. Kumar and L.M. Patnaik, "Extended Hypercube: A Hierarchical Interconnection Network of Hypercube", *IEEE Trans on Parallel and Distrib. Systems*, vol. 3, no. 1, 1992, pp. 45-57.
- [4] N.F. Tzeng and S. Wei, "Enhanced Hypecube," *IEEE Trans. on Computers*, vol. 40, no. 3, 1991, pp. 284-294.
- [5] S.G. Ziavars, "A Versatile Family of Reduced Hypercube Interconnection Network", *IEEE Trans on Parallel and Distributed Systems*, vol. 5, no. 11, 1994, pp. 1210-1220.
- [6] V.K. Jain, T. Ghirmai, and S. Horiguchi, "TESH: A new hierarchical interconnection network for massively parallel computing", *IEICE Trans. on Inf. & Syst.*, vol. E80-D, no. 9, 1997, pp. 837-846.
- [7] V.K. Jain and S. Horiguchi, "VLSI Considerations for TESH: A New Hierarchical Interconnection Network for 3-D Integration", *IEEE Trans on VLSI Systems*, Vol. 6, no. 3, 1998, pp. 346–353.
- [8] Y. Miura, "Wormhole Routing for Hierarchical Interconnection Networks", *Ph.D. Dissertation*, School of Information Science, JAIST, 2002.
- [9] W.J. Dally, "performance Analysis of k -ary n -cube Interconnection Networks", *IEEE Trans. on Computers*, vol. 39, no. 6, 1990, pp. 775-785.
- [10] M.M. Hafizur Rahman, Y. Inoguchi, Y. Sato, and S. Horiguchi, "TTN: A High performance Hierarchical Interconnection Network for Massively Parallel Computers", *IEICE Trans. on Inf. & Syst.*, vol. E92-D, no. 5, 2009, pp.1062-1078.
- [11] M.M. Hafizur Rahman, Xiaohong Jiang, and Faiz Al Faisal, "A Cost Efficient Hierarchical Interconnection Network", *Submitted in the Journal of Parallel Computing*, Elsevier, Holland, 2009.