www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume - 3 Issue -7 July, 2014 Page No. .7388-7393

Routing and Sorting on OTIS-Hyper Hexa-Cell

Abdul Hannan Akhtar[1], Keny T. Lucas[2]

[1]Government Polytechnic,
Ranchi, Dist. Ranchi, Jharkhand, PIN – 834001, India
abdulhannan.akhtar@yahoo.com

[2]Xavier Institute of Polytechnic and Technology,
Namkum, Dist. Ranchi, Jharkhand, PIN - 834010, India
kenny.lucas@gmail.com

Abstract: ***In the recent years, the Optical Transpose Interconnection System (OTIS) has attracted researchers for solving computational and communicational intensive problems. OTIS is a hybrid interconnection network that exploits the advantages of electronic link as well as optical links for connecting processors in the network. Many variants of OTIS model and several parallel algorithms for different problems have been proposed on those models. In this paper, we have proposed shortest path routing and sparse enumeration sort on OTIS hyper hexa-cell. In the sparse enumeration sort, the number of keys to be sorted is p^x, for some constant $x \leq ½$ and for our proposed algorithm $x=2$. The proposed shortest path routing on OTIS-HHC is optimal as the number of steps required is equal to its diameter $(2d_h+3)$. The time complexity of algorithm proposed for parallel sparse enumeration sort is $4(d_h+1)$ electronic moves + 4 OTIS moves for case I and $4(d_h+1)$ electronic moves + 3 OTIS moves for case II.***

Keywords: parallel algorithm, interconnection network, OTIS Hyper hexa-cell, routing, enumeration sort, time complexity.

1. Introduction

Optical Transpose Interconnection System (OTIS) has become one of the most popular interconnection networks in the recent year for developing parallel algorithm for solving computation as well as communication intensive problems. An OTIS is a hybrid interconnection Network that exploits electronic and optical links for connecting processing nodes of a parallel architecture [1], [2], [3]. The electronic links are used to connect the processors that are in a range of few millimeters and these processors can be fabricated within a chip. The free space optical links are used to provide the connectivity between the processors of different groups. In an OTIS model, the processors are organized in groups. Each group within any OTIS model contains equal number of processors and the number of groups may or may not be equal to the number of processors in each group. However, if the number of groups and the number of processors in each group is same, then the bandwidth of the OTIS systems can be maximized and the power consumption can be minimized [4], [5], [6]. The interconnection pattern of any OTIS model is decided by the pattern of each group. If the interconnection pattern G is implemented to connect the processors within the group, then the overall architecture of the OTIS architecture is termed as OTIS-G. In the recent years, many researchers have exploited OTIS architecture to propose parallel algorithms for solving scientific and engineering problems. A rich literature of work on OTIS models and optical networks is available such as basic operations [7], matrix multiplication [8], [9], BPC permutation [10], sorting 11], [12], [13], [14], [15], [16] routing [11], [13], image processing [17], conflict graph construction[18] load balancing [19], [20], polynomial interpolation [21], polynomial root finding [21], [22], prefix computation [23], gossiping [24, 25, 26].

In this paper, we have proposed algorithms for shortest path routing and sparse enumeration sort on the newly proposed architecture called OTIS-hyper hexa-cell (OTIS-HHC) [27]. In sparse enumeration sort the number of keys to be sorted is much less than the network size. Here, the number of keys is assumed to be p^β for some constant $\beta \leq ½$, where p is the network size. This assumption has also been followed in [28], [29], [13], [16]. The proposed algorithm for shortest path routing is optimal as the number of steps required is equal to the diameter of OTIS-HHC. The parallel algorithm for sparse enumeration sort proposed in this paper has two versions. For case I, it requires $4(d_h+1)$ electronic moves + 04 OTIS moves and for case II, $4(d_h+1)$ electronic moves + 03 OTIS moves are required. The same algorithm has been proposed for OTIS-MOT [13] and OTIS k-ary n-cube [16].

The paper is organized as follows: the topology of the OTIS-HHC is presented in section 2. In section 3, we have presented the proposed algorithm for shortest path routing and sparse enumeration sort followed by conclusion in section 4.

2. Topology of OTIS-hyper hexa-cell

The topological structure of an OTIS-HHC is based on the structure of hypercube, hyper hexa-cell (HHC) and OTIS topologies [27]. The topology of each group in an OTIS-HHC is a hyper hexacell. A d_h-dimensional HHC is an undirected graph and is formed by replacing 2d nodes of a hypercube by replacing one-dimensional HHC. Each d_h-dimensional HHC constitutes a hypercube of dimension d_h+1. Various topological properties have been discussed in detail in [27]. The diameter of OHHC is $(2d_h+3)$. The maximum and minimum node

degrees are (d_h+3) and (d_h+2) respectively. In an OTIS-HHC, the number of groups can be equal to the number of processors within each group or the number of groups can be half the number of processors within each group. The network size is $(6\times2^{dh-1})^2$ for G=P and $(6\times2^{dh-1})^2/2$ for G=P/2. The bisection width of an OTIS-HHC is $((6\times2^{dh-1})/2)^2$ for G=P and $((6\times2^{dh-1})/4)^2$ for G=P/2. The topology of one-dimensional OTIS-HHC is shown in Fig. 1.

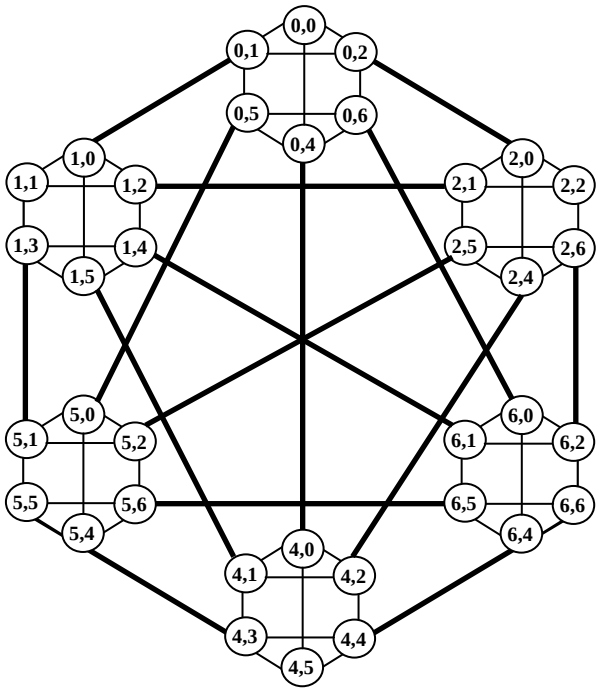

Figure 1. Topology of one-dimensional OTIS-HHC

3. Proposed Algorithms for Shortest Path Routing and Sparse Enumeration Sort

3.1 Shortest Path Routing

In an OTIS-HHC, the processors can be denoted by combining the group labels and node labels, i.e., <*u*, *v*> where *u* and *v* are the binary representations of group and node labels. Any processor <*u*, *v*> is connected to processor <*u'*, *v'*> of the same group only when *u* and *u*' differ by just one bit or *v* and *v*' differ by one bit. The maximum distance between the processors within the group is just two electronic links [27]. To discuss the shortest path routing in an overall OTIS-HHC architecture, let the source and destination nodes be <000,000> and <011,110> respectively as shown in Fig. 2. As the inter-group processors are connected through optical transpose rule, first we need to reach the processor within the source group that connects the destination group. The traversal within the group will require change in the bits of the node label and group traversal can be achieved by interchanging the node labels with group labels. First, we need to check the labels of the intermediate nodes with the destination node. This checking can be performed either from least significant bit (LSB) to most significant bit (MSB) or from most significant bit to least significant bit. We have used symbols "→" to represent the path through the electronic link and "➔" to show the optical path as shown below.

Case I: Checking the bits from LSB to MSB

<000,000> → <000,010> → <000,110> ➔ <110,000> → <110,010> → <110,110>. This requires 4 electronic links and one OTIS link.

Case II: Checking the bits from MSB to LSB

<000,000> → <000,100> → <000,110> ➔ <110,000> → <110,100> → <110,110>. This also requires 4 electronic links and one OTIS link.

Theorem 1: The shortest path algorithm on OTIS-HHC takes ($2d_h$+3) steps which is equal to the diameter of OTIS-HHC and hence it is optimal.

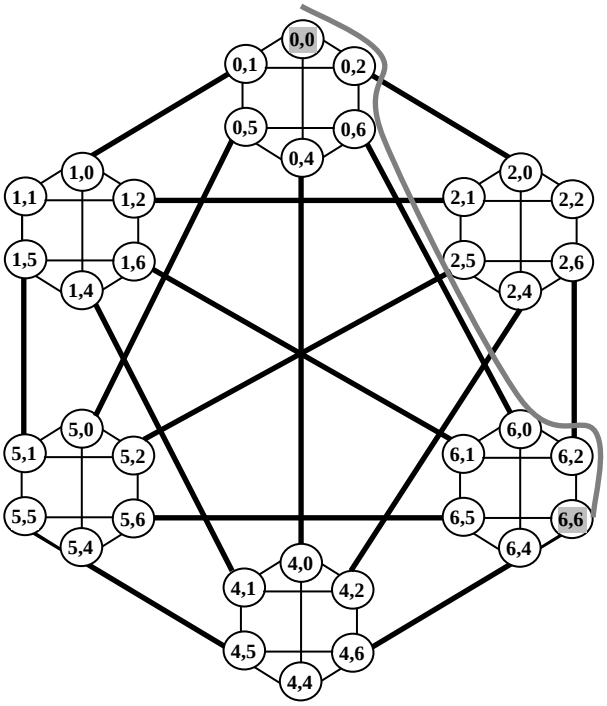

Figure 2. Source and destination nodes

3.2 Parallel Sparse Enumeration Sort

In this sorting, only *N* data elements are considered on a network size of N^2. Let us assume a data set {9, 2, 6, 4, 7, 8} for the sake of simplicity. In this algorithm, three registers *A*, *B*, and *C* are used. Registers *A* and *B* are used for storing the keys for comparison and *C* stores the rank of respective key in a processing node. We assume two cases of data initialization. In first case, initial data elements are stored in all processing nodes group G_0. In the second case, the data elements are initially stored in the node P_0 of all the groups. To analyze the time complexity of proposed algorithm for enumeration sort, we will count the number of data movements through electronic links as well as through optical links. We have also shown the illustrations after each step for this algorithm.

Algorithm Enum-Sort

Case I: all the data elements are populated in G_0

Step 1: Data Initialization

/* For all processors in G_0, do in parallel */

Initialize the data elements in all the processors

Step 2: /* For all processors in G_0, do in parallel */
$C := B$

Table 1: Contents of Registers after Step 2

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	9	2	6	4	7	8
	C	9	2	6	4	7	8
	A	-	-	-	-	-	-
G_1	B	-	-	-	-	-	-
	C	-	-	-	-	-	-
	A	-	-	-	-	-	-
G_2	B	-	-	-	-	-	-
	C	-	-	-	-	-	-
	A	-	-	-	-	-	-
G_3	B	-	-	-	-	-	-
	C	-	-	-	-	-	-
	A	-	-	-	-	-	-
G_4	B	-	-	-	-	-	-
	C	-	-	-	-	-	-
	A	-	-	-	-	-	-
G_5	B	-	-	-	-	-	-
	C	-	-	-	-	-	-
	A	-	-	-	-	-	-

Step 3: /* For all processors in G_0, do in parallel */
Perform OTIS move on Registers *B* and *C*

Table 2: Contents of Registers after Step 3

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	9	-	-	-	-	-
	C	9	-	-	-	-	-
	A	-	-	-	-	-	-
G_1	B	2	-	-	-	-	-
	C	2	-	-	-	-	-
	A	-	-	-	-	-	-
G_2	B	6	-	-	-	-	-
	C	6	-	-	-	-	-
	A	-	-	-	-	-	-
G_3	B	4	-	-	-	-	-
	C	4	-	-	-	-	-
	A	-	-	-	-	-	-
G_4	B	7	-	-	-	-	-
	C	7	-	-	-	-	-
	A	-	-	-	-	-	-
G_5	B	8	-	-	-	-	-
	C	8	-	-	-	-	-
	A	-	-	-	-	-	-

Step 4. /* For all Groups, do in parallel */
Perform Local Broadcast on Registers *B* and *C*

Table 3: Contents of Registers after Step 4

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	9	9	9	9	9	9
	C	9	9	9	9	9	9
	A	-	-	-	-	-	-
G_1	B	2	2	2	2	2	2
	C	2	2	2	2	2	2
	A	-	-	-	-	-	-
G_2	B	6	6	6	6	6	6
	C	6	6	6	6	6	6
	A	-	-	-	-	-	-
G_3	B	4	4	4	4	4	4
	C	4	4	4	4	4	4
	A	-	-	-	-	-	-
G_4	B	7	7	7	7	7	7
	C	7	7	7	7	7	7
	A	-	-	-	-	-	-
G_5	B	8	8	8	8	8	8
	C	8	8	8	8	8	8
	A	-	-	-	-	-	-

Step 5: /* For all Groups, do in parallel */
Perform OTIS move on *C*

Table 4: Contents of Registers after Step 5

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	9	9	9	9	9	9
	C	9	2	6	4	7	8
	A	-	-	-	-	-	-
G_1	B	2	2	2	2	2	2
	C	9	2	6	4	7	8
	A	-	-	-	-	-	-
G_2	B	6	6	6	6	6	6
	C	9	2	6	4	7	8
	A	-	-	-	-	-	-
G_3	B	4	4	4	4	4	4
	C	9	2	6	4	7	8
	A	-	-	-	-	-	-
G_4	B	7	7	7	7	7	7
	C	9	2	6	4	7	8
	A	-	-	-	-	-	-
G_5	B	8	8	8	8	8	8
	C	9	2	6	4	7	8
	A	-	-	-	-	-	-

Step 6: /* For all Groups, do in parallel */
/* For all processors, do in parallel */
If $B \geq C$ then
$A := A + 1$
Else
$A := A + 0$

Step 7. /* For all Groups, do in parallel */

Sum of the contents of Register *A* of all the processors within the group and broadcast it within the group

Table 5: Content of Registers after Step 6

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	9	9	9	9	9	9
	C	9	2	6	4	7	8
	A	1	1	1	1	1	1
G_1	B	2	2	2	2	2	2
	C	9	2	6	4	7	8
	A	0	1	0	0	0	0
G_2	B	6	6	6	6	6	6
	C	9	2	6	4	7	8
	A	0	1	1	1	0	0

G_3	B	4	4	4	4	4	4
	C	9	2	6	4	7	8
	A	0	1	0	1	0	0
G_4	B	7	7	7	7	7	7
	C	9	2	6	4	7	8
	A	0	1	1	1	1	0
G_5	B	8	8	8	8	8	8
	C	9	2	6	4	7	8
	A	0	1	1	1	1	1

Table 6: Content of Registers after Step 7

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	9	9	9	9	9	9
	C	9	2	6	4	7	8
	A	1	1	1	1	1	1
G_1	B	2	2	2	2	2	2
	C	9	2	6	4	7	8
	A	0	1	0	0	0	0
G_2	B	6	6	6	6	6	6
	C	9	2	6	4	7	8
	A	0	1	1	1	0	0
G_3	B	4	4	4	4	4	4
	C	9	2	6	4	7	8
	A	0	1	0	1	0	0
G_4	B	7	7	7	7	7	7
	C	9	2	6	4	7	8
	A	0	1	1	1	1	0
G_5	B	8	8	8	8	8	8
	C	9	2	6	4	7	8
	A	0	1	1	1	1	1

Step 8: /* For all Groups, do in parallel */
/* For all processors, do in parallel */

The content of Register *A* with rank *R* is to be moved to G_{R-1} through optical links using transpose rule and perform local broadcast on *A*

Step 9: /* For all Groups, do in parallel */
/* For all processors, do in parallel */

Perform OTIS move on Registers A and B where A holds the rank of B and the final result is shown in Table 9.

Time complexity: Steps 3, 5, 8 and 9, each takes one OTIS move. Steps 4 and 8, each needs (d_h+1) electronic moves whereas step 7 requires 2(d_h+1) electronic moves. Thus the overall time complexity can be given as 4(d_h+1) electronic + 04 OTIS moves.

Table 7: Processors sending data through optical link

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	9	9	9	9	9	**9**
	C	9	2	6	4	7	8
	A	6	6	6	6	6	**6**
G_1	B	**2**	2	2	2	2	2
	C	9	2	6	4	7	8
	A	**1**	1	1	1	1	1
G_2	B	6	6	**6**	6	6	6
	C	9	2	6	4	7	8
	A	3	3	**3**	3	3	3
G_3	B	4	**4**	4	4	4	4
	C	9	2	6	4	7	8
	A	0	1	1	**1**	1	1
G_4	B	7	7	7	7	**7**	7
	C	9	2	6	4	7	8
	A	4	4	4	4	**4**	4
G_5	B	8	8	8	8	**8**	8
	C	9	2	6	4	7	8
	A	5	5	5	5	**5**	5

Table 8: After OTIS move

		P_0	P_1	P_2	P_3	P_4	P_5
G_0	B	2	2	2	2	2	2
	C	9	2	6	4	7	8
	A	1	1	1	1	1	1
G_1	B	4	4	4	4	4	4
	C	9	2	6	4	7	8
	A	2	2	2	2	2	2
G_2	B	6	6	6	6	6	6
	C	9	2	6	4	7	8
	A	3	3	3	3	3	3
G_3	B	7	7	7	7	7	7
	C	9	2	6	4	7	8
	A	4	4	4	4	4	4
G_4	B	8	8	8	8	8	8
	C	9	2	6	4	7	8
	A	5	5	5	5	5	5
G_5	B	9	9	9	9	9	9
	C	9	2	6	4	7	8
	A	6	6	6	6	6	6

Case 2: The data elements are populated in P_0 of all the groups

Step 1: /* For all groups, do in parallel */
Initialize the data elements in P_0 of all the groups

Step 2: /* for all groups, for P_0 do in parallel */
C:=*B*
Step 3: Step 4 of Case I
Step 4: Step 5 of Case I
Step 5: Step 6 of Case I
Step 6: Step 7 of Case I
Step 7: Step 8 of Case I
Step 8: Step 9 of Case I

Table 9: Final sorted key elements

		p=0	p=1	p=2	p=3	p=4	p=5
G_0	B	**2**	**4**	**6**	**7**	**8**	**9**
	C	9	2	6	4	7	8
	A	**1**	**2**	**3**	**4**	**5**	**6**
G_1	B	**2**	**4**	**6**	**7**	**8**	**9**
	C	9	2	6	4	7	8
	A	**1**	**2**	**3**	**4**	**5**	**6**
G_2	B	**2**	**4**	**6**	**7**	**8**	**9**
	C	9	2	6	4	7	8
	A	**1**	**2**	**3**	**4**	**5**	**6**
G_3	B	**2**	**4**	**6**	**7**	**8**	**9**
	C	9	2	6	4	7	8
	A	**1**	**2**	**3**	**4**	**5**	**6**
G_4	B	**2**	**4**	**6**	**7**	**8**	**9**
	C	9	2	6	4	7	8
	A	**1**	**2**	**3**	**4**	**5**	**6**
G_5	B	**2**	**4**	**6**	**7**	**8**	**9**

	C A	9 **1**	2 **2**	6 **3**	4 **4**	7 **5**	8 **6**

Time Complexity: In the later approach, the algorithms requires $4(d_h+1)$ electronic moves + 03 OTIS moves.

The AT cost of our proposed algorithm for sparse enumeration sort for OTIS-HHC is minimum compared to algorithms proposed in [13] and [16] as shown in the table below:

Table 10: Comparision of AT costs

Architecture	Network Size	Electronic move	OTIS move
OTIS-MOT	81	2054.111	243
OTIS k-ary n-cube	36	432	108
OTIS-HHC	81	1296	162

4. Conclusion

The shortest path algorithm proposed for OTIS-HHC is optimal as the number of steps required is equal to the diameter of the network. In this paper, two approaches of data initialization have been considered for the sparse enumeration sort. The time complexity of the algorithm for the first case is $4(d_h+1)$ electronic moves + 04 OTIS moves and for the second case it is $4(d_h+1)$ electronic moves + 03 OTIS moves.

References

[1] G. Marsden, P. Marchand, P. Harvey and S. Esener, "Optical Transpose Interconnection System Architecture", Optics Letters, XVIII(13), pp 1083-1085, 1993.

[2] F. Kiamilev, P. Marchand, A. Krishnamoorthy, S. Esener, S. Lee, "Performance Comparison between Optoelectronic and VLSI Multistage Interconnection Networks", Journal of Light Wave Technology, 9(12), pp 1674 – 1692, 1991.

[3] F. Zane, P. Marchand, R. Paturi, S. Esener, "Scalable Network Architecture using the Optical Transpose Interconnection System (OTIS)", In Proceedings of the International Conference on Massively Parallel Processing using Optical Interconnections (MPPOI' 96), San Antonio, Texas, pp 114-121, 1996.

[4] F. Kiamilev, P. Marchand, A. Krishnamoorthy, S. Esener, S. Lee, "Performance Comparison between Optoelectronic and VLSI Multistage Interconnection Networks", Journal of Light Wave Technology, 9(12), pp 1674-1692, 1991.

[5] A. Krishnamoorthy, P. Marchand, F. Kiamilev, S. Esener, "Grain-size Consideration for Optoelectronic Multistage Interconnection Networks", Applied optics, XXXI(26), pp 5480-5507, 1992.

[6] M. Feldman, S. Esener, C. Guest, S. Lee, "Comparison between Electrical and Free-space Optical Interconnects based on Power and Speed Consideration", Applied Optics, XXVII(9), pp 1742-1751, 1990.

[7] C. –F. Wang and S. Sahni, "Basic Operations on OTIS-Mesh Optoelectronic Computer", IEEE Transaction on Parallel and Distributed Systems, IX(12), pp 1226-1236, 1998.

[8] C. –F. Wang and S. Sahni, 2001, "Matrix Multiplication on the OTIS-Mesh Optoelectronic Computer", IEEE Transactions on Computers, XXXXX(7), pp 635-646, 2001.

[9] Mallick D. K. and Jana P. K., "Matrix Multiplication on OTIS *k*-Ary 2-Cube Network", In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, pp 224-228, 2008.

[10] S. Sahni and C. -F. Wang, "BPC Permutations on the OTIS Hypercube Optoelectronic Computer", Informatica, XXII, pp 263-269, 1998.

[11] S. Rajasekaran and S. Sahni, "Randomized Routing, Selection and Sorting on OTIS-Mesh", IEEE Transaction on Parallel and Distributed Systems, IX(9), pp 833-840, 1998.

[12] A. Osterloh, "Sorting on OTIS-Mesh", In Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS 2000), pp 269-274, 2000.

[13] K. T. Lucas and P. K. Jana, "Sorting and Routing on OTIS-Mesh of Trees", Parallel Processing Letters, XX(2), pp 145-154, 2010.

[14] K. T. Lucas, "Parallel Enumeration Sort on OTIS-Hypercube", In Proceedings of the IC3, pp 21-31.

[15] K. T. Lucas, "Parallel algorithm for Sorting on OTIS-Ring Computer", In Proceedings of the Bangalore Annual Compute Conference (COMPUTE 2009), pp 1-5.

[16] A. H. Akhtar and K. T. Lucas, "Enumeration Sort on OTIS k-ary n-cube Architecture", International Journal of Engineering and Computer Science, III(7), pp 7173-7176, 2014.

[17] C. –F. Wang and S. Sahni, "Image Processing on the OTIS-Mesh Optoelectronic Computer", IEEE Transactions on Parallel and Distributed Systems, XI(2), pp 97-109, 2000.

[18] K. T. Lucas, D. K. Mallick, P. K. Jana, "Parallel Algorithms for the Conflict Graph on the OTIS Triangular Array", In Proceedings of the International Conference on Distributed Computing and Networking, Springer-Verlag Berlin Heidelberg, pp 274-279, 2008.

[19] B. Mahafzah and B. Jaradat, "The Load Balancing Problem in OTIS-hypercube Interconnection Networks", Journal of Supercomputing, XXXXVI(3), pp 276-297, 2008.

[20] C. Zhao, W Xiao, B. Parhami, "Load-balancing on a Swapped or OTIS Networks", Journal of Parallel and Distributed Computing, XXXXXXIX(4), pp 389-399, 2009.

[21] P. K. Jana, "Polynomial Interpolation and Polynomial Root Finding on OTIS-Mesh", Parallel Computing, XXXII(4), pp 301-312, 2006.

[22] K. T. Lucas and P. K. Jana, "Parallel Algorithms for Finding Polynomial Roots on OTIS-Torus", Journal of Supercomputing, XXXXIV(2), pp 139-153, 2010.

[23] K. T. Lucas, "Parallel Algorithm for Prefix Computation on OTIS *k*-ary *n*-cube Parallel Computer", In Proceedings of the International Journal of Recent Trend in Engineering, I(1), pp 560- 562, 2009.

[24] K. T. Lucas, "The Gossiping on OTIS-Hypercube Optoelectronic Parallel Computer", In Proceedings of the International Conference on Parallel and Distributed Techniques and Applications, Las Vegas, Nevada, USA, pp 185-189, 2007.

[25] T. F. Gonzalez, "An Efficient Algorithm for Gossiping in the Multicasting Communication Environment", IEEE

Transactions on Parallel and Distributed Systems, XIV(7), pp 701-708, 2003.
[26] Lucas K. T., "The Gossiping Algorithm for OTIS *k*-Ary *n*-Cube Parallel Computer", In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, pp 253-257, 2008.
[27] B. A. Mahafzah, A. Sleit, N. A. Hamad, E. F. Ahmad, T. M. Abu-Kabeer, "The OTIS Hyper Hexa-cell Optoelectronic Architecture", Computing, XXXXXXXXXIV, pp 411-432, 2012.
[28] E. Horowitz, S. Sahni and S. Rajasekaran, Fundamentals of Computer Algorithms, Galgotia Publications Pvt. Ltd,, New Delhi, 2002.
[29] S. G Akl, Parallel Sorting Algorithm, Academic Press, Orlando, FI, 1985.