



An Efficient Routing Implementation for Irregular Networks

By Chand Mal Samota, Naveen Choudhary & Dharm Singh

Maharana Pratap University of Agriculture and Technology, India

Abstract- With the recent advancements in multi-core era, workstation clusters have emerged as a cost-effective approach to build a network of workstations (NOWs). NOWs connect the small groups of processors to a network of switching elements that form irregular topologies. Designing an efficient routing and a deadlock avoidance algorithm for irregular networks is quite complicated in terms of latency and area of the routing tables, thus impractical for scalability of On Chip Networks. Many deadlock free routing mechanisms have been proposed for regular networks, but they cannot be employed in irregular networks. In this paper a new methodology has been proposed for efficient routing scheme, called LBDR-UD, which save the average 64.59% routing tables in the switch for irregular networks as compare to up*/down* routing. The Basic concept of routing scheme is combination of up*/down* and Logic Based Distributed Routing. By simulation, it has been shown that the LBDR-UD is deadlock free and adaptive to all dynamic network traffic conditions.

Keywords: up*/down*, routing, LBDR, irregular networks, IBDR-UD.

GJCST-E Classification : C.2.2



Strictly as per the compliance and regulations of:



An Efficient Routing Implementation for Irregular Networks

Chand Mal Samota^α, Naveen Choudhary^σ & Dharm Singh^ρ

Abstract- With the recent advancements in multi-core era, workstation clusters have emerged as a cost-effective approach to build a network of workstations (NOWs). NOWs connect the small groups of processors to a network of switching elements that form irregular topologies. Designing an efficient routing and a deadlock avoidance algorithm for irregular networks is quite complicated in terms of latency and area of the routing tables, thus impractical for scalability of On Chip Networks. Many deadlock free routing mechanisms have been proposed for regular networks, but they cannot be employed in irregular networks. In this paper a new methodology has been proposed for efficient routing scheme, called LBDR-UD, which save the average 64.59% routing tables in the switch for irregular networks as compare to up*/down* routing. The Basic concept of routing scheme is combination of up*/down* and Logic Based Distributed Routing. By simulation, it has been shown that the LBDR-UD is deadlock free and adaptive to all dynamic network traffic conditions.

Keywords: up*/down*, routing, LBDR, irregular networks, IBDR-UD.

I. INTRODUCTION

Multi Processor-SoC (MPSoC) and Chip-Multi Processors (CMP) achieve high performance with interconnection networks that give low-latency, high-bandwidth inter-processor communication [7]. Most of these multi-core system use regular topologies (such as torus, hypercube and mesh) to link their switch components. For packet transmission, many routing schemes have been design to provide an efficient and deadlock free path [1, 2, 4, 5, 8, 10]. Routing algorithm (RA) decides the path for the packet from source to destination. Two type of RA i.e. distributed and source routing are used for regular and irregular NoC networks [8]. Source routing compute the whole routing path at the source and computed path stored in the packet header, while in distributed routing each router receives a packet; all computations are performed at the switch level, without storing whole path in packet header and decides the output direction to send it. In recent years, several routing schemes have been proposed for application specific networks (i.e. Irregular networks) [6, 9, 11, 12, 13, 14, 15, 16]. These

Author α σ ρ: Department of Computer Science and Engineering, College of Technology and Engineering, Maharana Pratap University of Agriculture and Technology, Udaipur, Rajasthan, India.
e-mails: cmsamota@gmail.com, naveenc121@yahoo.com & singhdharm@hotmail.com

schemes are able to route packets in different network topologies and achieves livelock and deadlock freedom. To deal with irregular topologies, table based approaches were proposed. In this scenario, at each switch that stores a table, for each end-node, the output port that must be used. Using this approach higher adaptivity is achieved and several outputs are stored in each table. The main benefit of this type of routing is that any topology and any routing algorithm can be used; it also supports fault-tolerant routing algorithms. With such routing approaches, the size of routing table increases proportionally with the size of the network at each switch. Hence, the implementation becomes comparatively complex for the communication switch.

In this paper a efficient routing algorithm is proposed for irregular networks. The aim is to develop a distributed routing on every switch and remove the tables, the routing decision is made quickly and low latency for packets sending from source to destination. The wormhole switching technique is used in our interconnection networks. The routing algorithm is based on combination of up*/down* and LBDR, and hence called LBDR Up*/Down* (LBDR-UD). [12] up*/down* routing is distributive table-based routing algorithm used in the irregular network and [7] LBDR is a table-less routing implementation technique for regular and irregular networks (generated from the mesh).

II. LBDR-UD ROUTING FOR IRREGULAR NETWORKS

We start the description with the basic mechanism required at every switch to deal with the irregular networks.

a) Methodology

In order to enable an efficient routing implementation for irregular networks using minimum logic, an example network for irregular Network-on-Chip with core and channels are placed according to the application condition as shown in Fig. 1.

The proposed methodology is based on two assumptions:

- i. The interconnection network between switches can be modeled by a multigraph $G(N, C)$, where N is the group of switches and C is the group of bidirectional links between the switches.

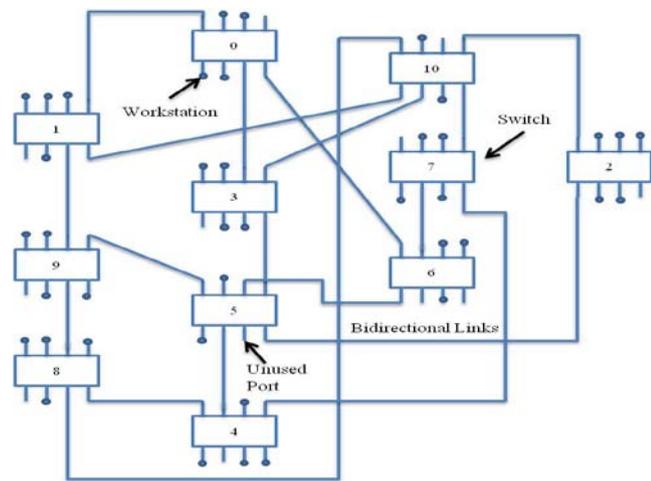
ii. For each irregular networks, applied routing algorithm must follow some restrictions. These restrictions are as follows:

a. *Deadlock-freedom*

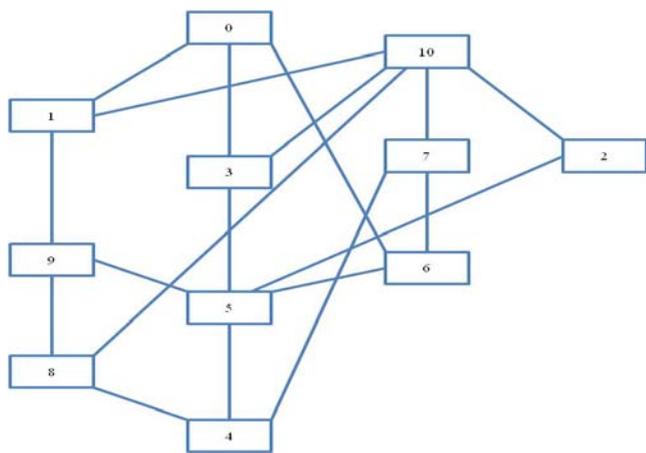
The routing algorithm must guarantee that the transmitted messages are received at destination and prevent the deadlock scenarios.

b. *Connectivity*

It is essential that the routing algorithm should be capable to offer at least one route between two end nodes.



(a)



(b)

Figure 1 : An example of custom NoC with irregular topology (a) irregular networks (b) corresponding topology graph

Routing is based on a direction assignment to the operational links, including the ones that do not belong to the tree. In particular, the “up” direction of each link is defined as: 1) A link leading to a parent node in the spanning tree; 2) A link leading to a lower ID, if both are at the same tree level. The “down” direction is along the reverse direction of the “up” direction as shown in Fig. 2.

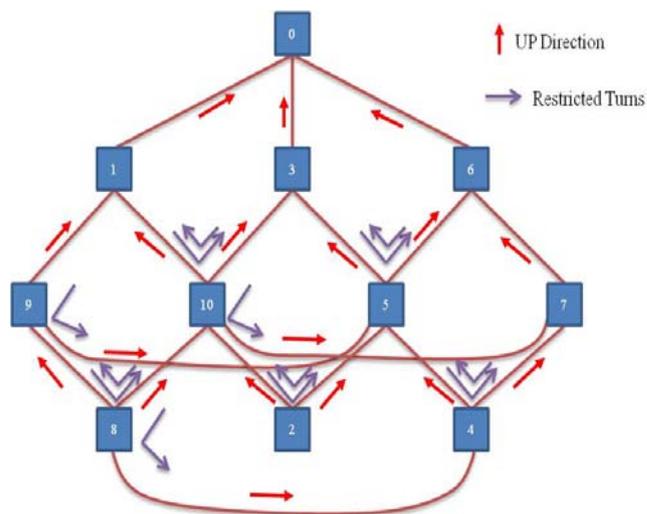


Figure 2 : Routing restrictions based on LBDR-UD algorithm

The routing restrictions for the LBDR-UD routing algorithm are shown in Fig.2. A routing restriction is defined between two successive channels. The LBDR-UD algorithm prohibits messages from taking “down-up” transitions. The transitions are allowed in opposite direction by algorithm, thus routing restrictions does not exist for “up” link to “up” and “down” links. Similarly, no routing restrictions are applicable from “down” link to “down” link transitions.

b) *Configuration bits*

Configuration bits are in two sets: connectivity bits and routing bits. Routing bits specify which routing options can be used, whereas connectivity bits specify connectivity with neighbors.

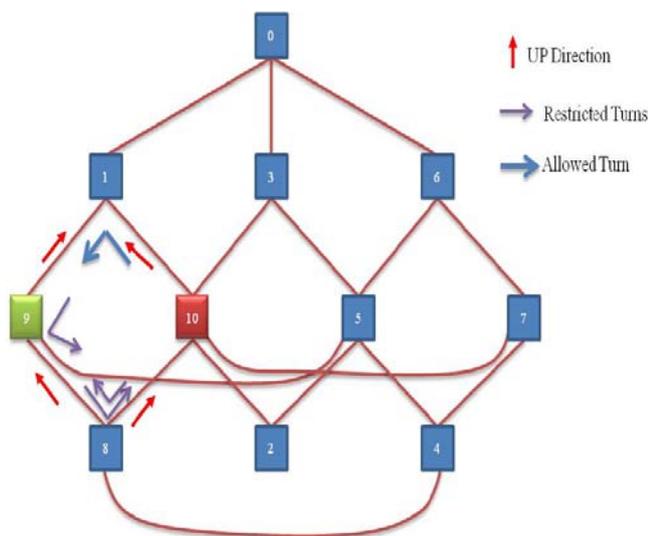
i. R_{UU} and R_{UD}

These bits indicate that the message can take the “up” direction from the current router and from the next router the message can be transmitted in the “up” direction or the “down” direction respectively.

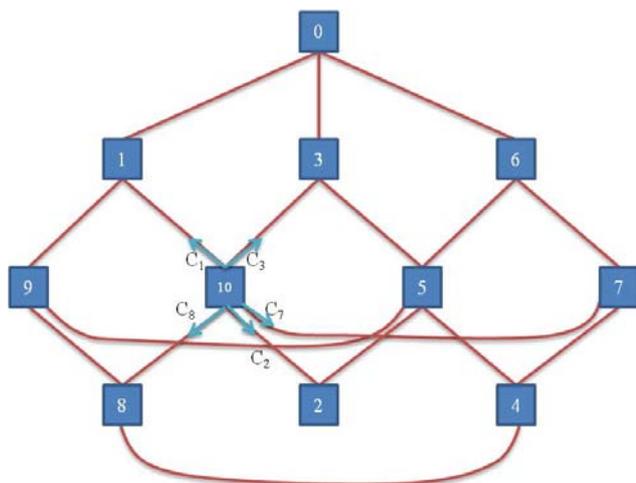
ii. R_{DD} and R_{DU}

These bits indicate that the messages can take the “down” direction from the current router and from the next router the message can be transmitted in the “down” direction or the “up” direction respectively.

Note that the routing restrictions and routing bits are the opposite to each other. Fig 3(a) shows the restricted turns and allowed turns a message could take according to router 10 and its routing bits. Specifically, bit R_{UD} at router 10 is 1(Set) and shows that a message is routed to the “up” direction first and then to the “down” direction from the next router. Routing decision is taken again at the next corresponding router.



(a) Routing bits



For router 7, $C_1=C_3=C_8=C_7=C_2=1$

(b) Connectivity bits

Figure 3 : Configuration bits at router 7. LBDR-UD algorithm used

A C_y bit defines the connectivity bits for the y^{th} output port. For example, at router 10 if the C_1 bit is set, it implies that there is a neighbor router connected with 1. Fig 3(b) shows all connectivity bits for router 10.

Table 1 shows the bits computed for an irregular network using the LBDR-UD routing algorithm (connectivity bits can be seen in Fig. 3(b)). As shown in the figure, bit R_{DU} is set to zero, representing the “down” to “up” routing restrictions which “LBDR-UD” imposes. Bits R_{UU} and R_{DD} are all set except for those cases where the message would be routed out of the network (at the root router and leaf router). R_{UD} is set (“up” to “down” transitions are allowed) except for those cases where the message would be routed out of the network.

Finally, based on neighboring routers, connectivity bits are set.

Fig. 4 shows the algorithm in pseudo-code for the computation of configuration bits. Function check link shows that whether a link between current router and neighboring router exist or not. Based on the type of restriction the routing bit is set on behalf of its immediate neighbor router. In the LBDR-UD routing only single turn (RDU) is restricted and rest three turns (RUU, RDD and RUD) are allowed.

Input : Topology definition with location of routing restriction and existent links

Output : Configuration bits

Procedure :

//checking connectivity bits of current router

for each router

for (all neighbours of current router)

$C_n = 1$; //connectivity bit of neighbour

end for

for each router // checking restriction, defined as unidirectional

getRoutingRestriction(Curr_id);

end for

Figure 4 : Pseudo-code for the computation of configuration bits

The propose algorithm is flexible, simple and compact routing mechanism for unicast communication that eliminates the requirement for routing tables at every router. The routing logic is separated in two stages. The first stage computes the location of the relative destination router. A general representation of LBDR-UD routing is shown in Fig. 5 and Fig. 6 Initially a comparator module is used, which generates two control signals using three comparators. First comparator (CMP) is used to compare the current_id_level and Dest_id_level, and if the levels are equal then it compares using the Direct Connection CMP otherwise the Ancestor CMP is used. These signals, U1 and D1 indicate the relative direction of the destination in “up” direction and “down” direction respectively.

For example, in Fig. 2, if the current router is 7 and our destination is router 6, signal U1 would be set, because it is situated at the lower level in comparison with the current router. With the help of these connectivity (C_u) bits, routing (R_{UD}) bits and control signals, the LBDR-UD routing generates a set of routing

Table 1 : Connectivity bits for an irregular network, Routers are numbered row-wise. (See Fig. 3(b))

Router	C ₀	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	C ₇	C ₈	C ₉	C ₁₀
0	0	1	0	1	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	1	0	0	1
3	1	0	0	0	0	1	0	0	0	0	1
4	0	0	0	0	0	1	0	1	1	0	0
5	0	0	1	1	1	0	1	0	0	1	0
6	1	0	0	0	0	1	0	1	0	0	0
7	0	0	0	0	1	0	1	0	0	0	1
8	0	0	0	0	1	0	0	0	0	1	1
9	0	1	0	0	0	1	0	0	1	0	0
10	0	1	1	1	0	0	0	1	1	0	0

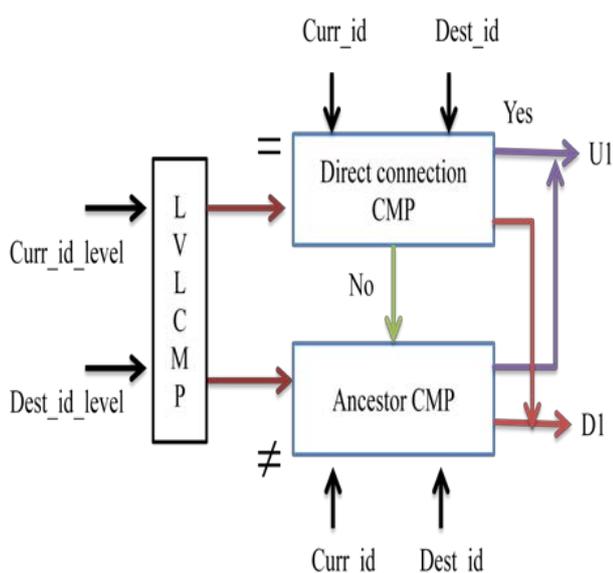


Figure 5 : Comparator module used in the first stage

decisions at the next stage. The second stage requires two or more logic units and each logic unit will correspond to one output port. Each output port can be implemented with only one inverters, one OR gate and three AND gates. We describe here the logic associated with the up and down output direction, as shown in figure 3.19.

When any of the falling out three conditions is fulfilled, then the incoming packet is headed in “up” output direction for routing. The “up” direction is not considered for routing the packet when none of the following conditions are fulfilled (additionally, the connectivity bit CU is examine in order to filter the up direction).

- The destination is directly connected with destination $U1 \times \overline{D1}$
- The destination is the ancestor of source router and the message can take the up direction at the lower level router through the up direction output port $U1 \times R_{UV}$.

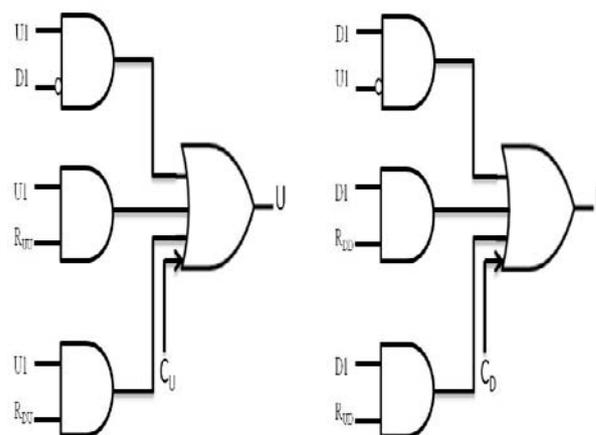


Figure 6 : LBDR-UD implementation, using logic gates.

- The destination is a descendant of source router and the transmitted message can be headed to the up direction at the upper level router via the down direction output port $U1 \times R_{DV}$.

As stated above, this logic provides support for nonminimal and minimal paths of the network, and produces a signal for each output port. When U1 and D1 signals are reset, then the C (Core) signal is set and the message is received at the final destination router.

III. RESULTS AND ANALYSIS

In the following, we provide performance evaluation of LBDR-UD and TABLE based (up*/down*) with different network size. Average memory, latency and throughput are calculated on 10 different scenario of each network size (Total 60 scenarios).

a) Effect of Network Size on Memory

Distributed Table based routing schemes have also been proposed to deal with irregular topologies and can be used in application-specific systems which facilitates the use of any topology with any routing algorithm. On small systems the hardware cost and power consumption related to the memories used to build routing tables is affordable, but as more and more cores are integrated on the chip, causing the system

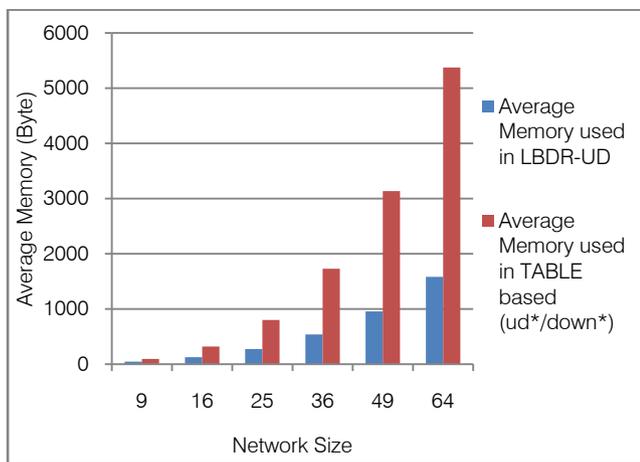


Figure 7 : Average Memory comparative result between LBDR-3D and Table based (up*/down*) routing with different network size

b) Effect of Network Size on Latency

Increase the network size also increases the packet transmission delay. Fig. 8 shows the average per flit latency of different network sizes. LBDR-UD performs the bit logic computation for route the message (packets). LBDR-UD and TABLE based (up*/down*) routing generate the approximately same average latency with respect to different network size but compare to Table based, LBDR-UD slight less as shown in fig. 8 Searching time for the corresponding entry in the routing table per router reduces for the LBDR-UD.

c) Effect of Network Size on Throughput

Fig. 9 shows the Average Throughput/packet with different network size. LBDR-UD same throughput as compare to TABLE based (up*/down*) routing.

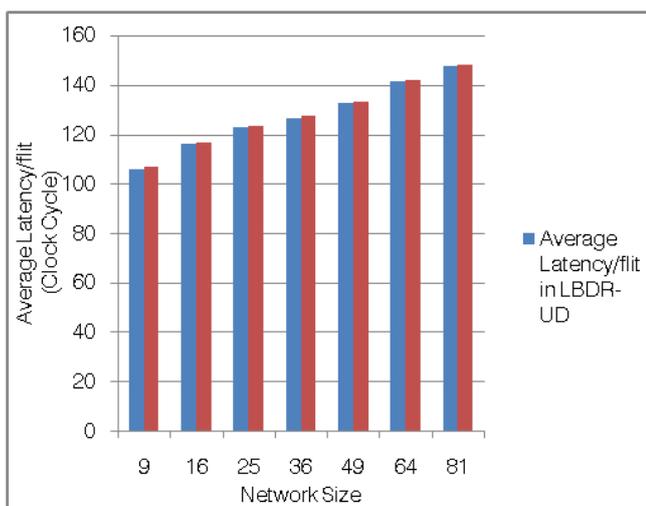


Figure 8 : Average Latency/flit comparative result between LBDR-3D and Table based (up*/down*) routing with different network size

Increases the network size the throughput/packet is same for both table based and LBDR-UD, because NoC is a delay model, all sending packets from source are received at destination. As shown fig. 9 LBDR-UD and TABLE routing throughput with respect to different network size.

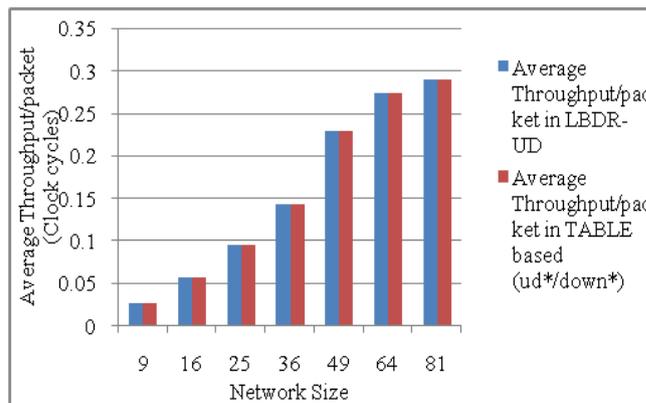


Figure 9 : Average Throughput/packet comparative result between LBDR-3D and Table based (up*/down*) routing with different network size

VI. CONCLUSION

It is extension of up*/down* routing in irregular networks using LBDR. LBDR-UD minimizes the memory space of the table based routing (up*/down*) in the NoC for irregular networks. On the behalf of simulation evaluation LBDR-UD save the average 64.59% tables and also efficient in terms of network average latency/flit and throughput/packet as compare to TABLE based (up*/down*) routing.

In application specific topologies (Irregular networks) LBDR also used for 3D irregular networks and this research will be extended by designing the efficient router architecture for NoC.

REFERENCES RÉFÉRENCES REFERENCIAS

1. Boppana, R. V. and Chalasani, S., 1993, A Comparison of Adaptive Wormhole Routing Algorithms, *Proceeding Intel Symposium on Computer Architecture*, pp. 351-360.
2. Chien A. and Kim, J. H., 1995, Planar-Adaptive Routing: Low-Cost Adaptive Networks for Multiprocessors, *Journal of ACM*, pp. 91-123.
3. Choudhary N., 2013, Principles of on-chip Interconnection Networks, Himanshu Publication.
4. Dally, W. J. and Seitz, C. L., 1987, Deadlock-Free Message Routing in Multiprocessor Interconnection Networks, *IEEE Trans. on Computers*, vol. 36, no. 5, pp. 547-553.
5. Duato, J., 1993, A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks, *IEEE Trans. On Parallel and Distributed Systems*, vol. 4, no. 12.

6. F. Silla and J. Duato, 1997, improving the efficiency of adaptive routing in networks with irregular topology, in Proceedings of the Conference on High Performance Computing.
7. Flich, J. and Duato, J., 2008, *Logic based distributed routing for NOCs*, *IEEE computer architecture letters*, vol. 7, no. 1.
8. Glass, C.J. and Ni, L. M., 1992, The Turn Model for Adaptive Routing, *Proceeding Intel. Symposium on Computer Architecture*, pp. 278-286.
9. Jouraku, A., Funahashi, A., Amano, H. and Koibuchi, M., 2001, L-turn routing: an adaptive routing in irregular networks, in *International Conference on Parallel Processing*, pp. 374-383.
10. Ni, L. M. and McKinley, P. K., 1993, A Survey of Wormhole Routing Techniques in Direct Networks," *IEEE on Computers*, vol. 26, pp. 62-76.
11. Sancho, J. C., Robles, A. and Duato, J., 2001, A new methodology to compute deadlock free routing tables for irregular networks", in Proceedings of the CANPC.
12. Schroeder, M.D., Birrell, A.D., Burrows, M., Murray, H., Needham, R.M., Rodeheffer, T.L., Satterthwaite, E.H. and Thacker, C.P., 1991, Autonet: A high-speed self-configuring local area network using point-to-point links,, *Journal on Selected Areas in Communications*, vol. 9.
13. Silla, F. and Duato, 2000, High-performance routing in networks of workstations with irregular topology, *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, pp. 699-719.
14. Silla, F., Malumbres, M. P., Robles, A., Lopez, P. and Duato J., 1997, Efficient adaptive routing in networks of workstations with irregular topology, in Proceedings of the Workshop on Communications and Architectural Support for Network-Based Parallel Computing, pp. 46-60.
15. Sun, Y.M., Yang, C.H., Chung, Y.C and Hang, T.Y., 2004, An efficient deadlock-free tree-based routing algorithm for irregular wormhole-routed networks based on turn model, in *International Conference on Parallel Processing*, vol. 1, pp. 343-352.
16. Wu, J. and Sheng, L., 1999, "Deadlock-free routing in irregular networks using prefix routing, DIMACS Tech. Rep. 99-19.

GLOBAL JOURNALS INC. (US) GUIDELINES HANDBOOK 2014

WWW.GLOBALJOURNALS.ORG