



GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY  
Volume 11 Issue 16 Version 1.0 September 2011  
Type: Double Blind Peer Reviewed International Research Journal  
Publisher: Global Journals Inc. (USA)  
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Energy Efficient Network Generation for Application Specific NoC

By Naveen Choudhary, M. S. Gaur , V. Laxmi

*Maharana Pratap University of Agriculture and Technology, Udaipur, Rajasthan, India*

**Abstract** - Networks-on-Chip is emerging as a communication platform for future complex SoC designs, composed of a large number of homogenous or heterogeneous processing resources. Most SoC platforms are customized to the domainspecific requirements of their applications, which communicate in a specific, mostly irregular way. The specific but often diverse communication requirements among cores of the SoC call for the design of application-specific network of SoC for improved performance in terms of communication energy, latency, and throughput. In this work, we propose a methodology for the design of customized irregular network architecture of SoC. The proposed method exploits priori knowledge of the application's communication characteristic to generate an energy optimized network and corresponding routing tables.

**Keywords** : SoC, on-chip networks, application specific NoC, design methodologies, Mesh topology, interconnection network.

**GJCST Classification** : H.2.8, D.2.9



*Strictly as per the compliance and regulations of:*



# Energy Efficient Network Generation for Application Specific NoC

Naveen Choudhary<sup>α</sup>, M. S. Gaur<sup>α</sup>, V. Laxmi<sup>β</sup>

**Abstract** - Networks-on-Chip is emerging as a communication platform for future complex SoC designs, composed of a large number of homogenous or heterogeneous processing resources. Most SoC platforms are customized to the domain-specific requirements of their applications, which communicate in a specific, mostly irregular way. The specific but often diverse communication requirements among cores of the SoC call for the design of application-specific network of SoC for improved performance in terms of communication energy, latency, and throughput. In this work, we propose a methodology for the design of customized irregular network architecture of SoC. The proposed method exploits prior knowledge of the application's communication characteristic to generate an energy optimized network and corresponding routing tables.

**Keywords** : SoC, on-chip networks, application specific NoC, design methodologies, Mesh topology, inter-connection network.

## 1. INTRODUCTION

The shrinking feature sizes in silicon technologies is making possible the integration of complex systems-on Chip (SoC), offering a remarkable amount of computational power. In order to address the design complexity and assist reuse, these systems are usually built from predesigned and preverified building blocks like general-purpose processor, a DSP, a memory subsystem, etc. Functionality of these systems is generally captured by a set of communicating tasks at a high level of abstraction. These tasks are mapped to computational resources which are interconnected by an underlying communication infrastructure.

NoC (Dally & Towles, 2001; Benini & DeMicheli, 2002; Kumar, Jantsch, Soininen, Forsell, Millberg, Oberg, Tiensyrja & Hemani, 2002; Ogras, Hu & Marculescu, 2005) has been recently proposed by academia and industry as the preferred choice for the communication infrastructure for the on-chip communication challenges of future SoC architectures. NoC is characterized by packet switching based communication mechanism that is enabled by on-chip routers. NoC architectures can be classified as custom

or regular based on their underlying communication infrastructure / topology. This communication infrastructure or topology impacts both performance and implementation costs of the system in terms of silicon area and energy consumption to a substantial extent.

A large number of NoC architectures have been proposed based on regular building patterns (Benini & DeMicheli, 2002; Kumar, Jantsch, Soininen, Forsell, Millberg, Oberg, Tiensyrja & Hemani, 2002; Natvig, 1997) like meshes, tori, k-ary n-cubes or fat trees for the implementation of on-chip networks to overcome conventional bus-based designs. However regular topologies may not be appropriate where communication requirement are not uniformly distributed across cores and links. Moreover most application specific SoCs are designed with static (or semi-static) mapping of tasks to processors or hardware cores and consequently the communication requirements of the SoC can be well characterized at design time. Therefore, the NoCs with irregular topology customized to the application's requirements is expected to be the preferred choice for application specific SoC platforms.

The routing function in NoC based systems is tightly coupled to the underlying topology defining the set of allowed paths on which packets may be sent from a sender to the destination core. The proper selection of the adequate topology and routing function form a key decision in the design of the overall NoC architecture. Conventionally, the proof of deadlock-freedom has mostly been carried out on the assumption of the regular topology (Dally & Seitz, 1987; Glass & Ni, 1992; Duato, Yalamanchili & Ni, 2003) and is far more complicated for NoC with underlying irregular topology. However in the NoC research domain some routing functions based on turn prohibition (Glass & Ni, 1992) methodology are proposed for irregular topology based NoCs such as prefix routing (Wu & Sheng, 1999), up\*/down\* (Schroeder et al. 1991), Left-Right (Jouraku, Funahashi, Amano & Koibuchi, 2001), L-turn (Jouraku, Funahashi, Amano & Koibuchi, 2001) and down/up (Sun, Yang, Chung & Hang, 2004).

In this paper, two genetic algorithm based heuristics are proposed for the design of energy efficient customized irregular topology Networks-on-Chip based on the applied routing function for application having IP cores with varying communication bandwidth

**Author <sup>α</sup>** : Department of Computer Science & Engineering, College of Technology and Engineering, Maharana Pratap University of Agriculture and Technology, Udaipur, Rajasthan, India.  
E-mail : naveenc121@yahoo.com

**Author <sup>α</sup>** : Department of Computer Engineering, MNIT, Jaipur, Rajasthan, India. E-mail : gaurms@gmail.com

**Author <sup>β</sup>** : Department of Computer Engineering, MNIT, Jaipur, Rajasthan, India. E-mail : vlaxmi@mnit.ac.in

requirements. The presented methodologies exploit the predefined communication requirements of the application to generate energy efficient customized NoC along with the routing tables for supporting deadlock free communication. It is worth mentioning here that the topology and routing table generation are tightly coupled aspects of the NoC design and therefore optimization of only one aspect or one after another may lead to suboptimal solutions. The paper is organized as follows. A brief account of related work is presented in Section II. Communication model and architecture for Irregular NoC are defined in Section III. The proposed genetic algorithm based energy efficient NoC design methodologies are presented in Section IV. The Genetic Algorithm used in the proposed methodologies is described in Section V. Experimental results are presented in Section VI followed by a brief conclusion in Section VII.

## II. RELATED WORK

Methods to collect and analyze traffic information that can be fed as input to the bus and NoC design processes have been presented in (Lahiri et al. 2004) and (Murali & De Micheli, 2005). Mappings of cores onto standard NoC topologies have been explored in (Murali & DeMicheli, 2004; Hansson et al. 2005; Hu & Marculescu, 2003; Murali et al. 2005). In (Murali & DeMicheli, 2004; Murali et al. 2005) a floorplanner is used during the mapping process to get area and wire-length estimates. These works only select from a library of standard topologies, and cannot generate a fully customized topology. In (Hansson et al. 2005), a unified approach to mapping, routing and resource reservation has been presented.

However, the work does not explore the topology design process. Important research in macro networks has considered the topology generation problem (Ravi et al. 2001). As the traffic patterns on these networks are difficult to predict most approaches are tree-based (like spanning or Steiner trees) and only ensure connectivity with node degree constraints. These techniques cannot be directly extended to address the NoC synthesis problem.

Application-specific custom topology design has been explored in (Pinto et al. 2003; Ho & Pinkston, 2003; Ahonen et al. 2004; Srinivasan et al. 2005). The works from (Pinto et al. 2003; Ho & Pinkston, 2003), do not consider the floorplanning information during the topology design process. In (Ahonen et al. 2004), a floorplanner is used during topology design to reduce power consumption on wires. It does not consider the area and power consumption of switches in the design. Also, the number and size of network partitions are manually fed. In (Srinivasan et al. 2005), a slicing tree based floorplanner is used during the topology design process. This work assumes that the switches are

located at the corners of the cores and does not consider the network components (switches, network interfaces) during the floorplanning process. Actual sizes of the cores in (Srinivasan et al. 2005; Srinivasan, & Chatha 2005) are considered only after generating their relative positions. The resulting floorplan can be extremely area inefficient when compared to the standard floorplanning process. In (Choudhary, N et al. 2010), a methodology to generate Bandwidth Aware NoC topology according to the application requirement is proposed. This methodology does floorplanning as the first step with high priority and later accomplishes topology generation with better traffic load distribution across the channels of the NoC leading to reduced congestion as well as hot spots in the topology. A range of issues in the design methods and tools for efficient synthesis of application specific Network-on-Chip interconnect for 3D SoC were addressed in (Seiculescu, Murali, Benini & De Micheli, 2009; Murali, Seiculescu, Benini & De Micheli, 2009).

In addition to the above, one of the major challenges for successful adoption of the Network-on-Chip paradigm is in reducing the energy consumed during the interaction between the IP cores. In (Hu & Marculescu, 2003; Hu & Marculescu, 2005), Hu and Marculescu have presented an energy-aware mapping algorithm to minimize the total communication energy cost for a 2-D mesh NoC architecture under real-time performance constraints. Similarly in (Choudhary, N., Gaur, M. S., Laxmi, V., Singh, V. (2010)) a deterministic methodology of order  $O(n^2)$  to generate energy efficient NoC topology is proposed. This methodology also does floorplanning as the first step with highest priority as in (Choudhary, N et al. 2010). However due to its deterministic nature the methodology is not capable to generate energy optimized NoC topology for all the given applications. The work in (Choudhary, N., Gaur, M. S., Laxmi, V., Singh, V. (2010)) is extended in this paper by incorporating a genetic algorithm based heuristic in the methodology for improved and optimized NoC topology generation. The methodology proposed in this work address the issue of topology/network design with deadlock free communication for application specific homogenous or heterogeneous NoC according to communication requirements. The proposed methodology accepts application's communication characteristics and floorplanning information as input. Therefore this methodology is especially suitable for applications where optimized placement of cores in chip layout during floorplanning based on metric such as area is done in advance with highest priority.

### III. IRREGULAR NOC COMMUNICATION MODEL ARCHITECTURE

The basic platform for the proposed methodology including the basic communication model assumed along with the associated NoC architecture and routing function are described in this section. The mapping of tasks in Task graphs (Hu & Marculescu, 2003; Dick, Rhodes & Wolf, 1998) to the actual physical IP cores in the NoC topology graph (NoC) can be done with the help of intermediate mapping to Core Graph as exhibited in Figure 1. The Core Graph and NoC topology graph can be defined as follows.

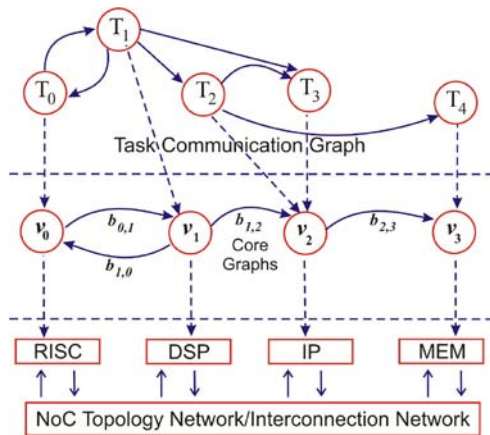


Fig. 1 : Application specific communication model for NoC.

**Definition 1 :** Core Graph is a directed graph,  $G(V, E)$  with each vertex  $v_i \in V$  representing an IP core and a directed edge  $e_{i,j} \in E$ , representing the communication between the cores  $v_i$  and  $v_j$ . The weight of the edge  $e_{i,j}$  denoted by  $b_{i,j}$ , represents the desired average bandwidth requirement of the communication from  $v_i$  and  $v_j$ .

**Definition 2 :** NoC topology graph is a directed graph  $N(U, F)$  with each vertex  $u_i \in U$  representing a node/tile in the topology and a directed edge  $f_{i,j} \in F$  represents direct communication channel between vertices  $u_i$  and  $u_j$ . Weight of the edge  $f_{i,j}$  denoted by  $b_{i,j}$  represents the available link/channel bandwidth across the edge  $f_{i,j}$ .

The energy model (Hu & Marculescu, 2003) for the regular Network-on-Chip can be defined as

$$E_{bit}(t_i, t_j) = n_{hops} \times Er_{bit} + (n_{hops} - 1) \times El_{bit}$$

Follows :

(1)

Where  $E_{bit}(t_i, t_j)$  is the average dynamic energy consumption for sending one bit of data from tile  $t_i$  to tile  $t_j$ ,  $n_{hops}$  is the number of routers the bit traverses from tile  $t_i$  to tile  $t_j$ ,  $Er_{bit}$  is the energy consumed by router for transporting one bit of data and  $El_{bit}$  is the energy consumed by link/channel for transporting one bit of

data. In case of Irregular NoC with unequal length channels for transporting data, the equation (1) can be modified as follows.

$$E_{bit}(t_i, t_j) = n_{hops} \times Er_{bit} + \sum_{k=1}^{n_{hops}-1} El_{bit}^k \quad (2)$$

Where the 2nd term of the summation in equation (2) represent the bit energy consumed by each channel in the route, the bit follows from communication source core to the intended destination cores.

For optimized chip layout, floorplanning according to desired metric like area can be done as a first step with the help of available floorplanning tools such as B\*-Trees (Chang, Chang, Wu & Wu, 2000; Lin & Chang, 2005).

The presented work uses the escape path based routing function as proposed by (Silla & Duato, 2000). To provide deadlock free communication in the NoC, the up\*/down\* routing (Schroeder et al. 1991; Silla et al. 1997) and Left-Right routing (Jouraku, Funahashi, Amano & Koibuchi, 2001) were used. These routing functions assign direction to the channels of the NoC with the help of a spanning tree of the give NoC topology.

In (Silla & Duato, 2000), a generic methodology for designing adaptive routing function for Irregular NoC was proposed. The proposed methodology allow messages to follow minimal paths, in most cases, reducing message latency and increasing network throughput (Duato, Yalamanchili & Ni 2003). Moreover the methodology enforces the deadlock free route to be followed only when the minimal path is occupied by other traffic/packet. This methodology assumes that all the physical channels in the NoC can be split into two virtual channels i.e. original virtual channel and the new virtual channel. Moreover the presence of a given deadlock free routing functions based on turn prohibition (Glass & Ni, 1992) for the given irregular NoC is also assumed. The methodology further proposes to extend the given routing function in such a way that newly injected messages can use new channels without any restriction as long as the original channels are used exactly in the same way as in the original routing function. In this paper original channels are made to use deadlock free paths based on up\*/down\* (Left-Right) deadlock free routing functions and new channels are allowed to follow the shortest available path to the destination. The modified routing function allows a packet arriving on a new channel following shortest path to be routed to any channel without any restrictions but preferably with higher priority to new channels as new channel assure shorter paths and higher adaptively (flexibility). If no new channels are available due to congestion, one of the original channels following up\*/down\* (Left-Right) must be provided. However, once a packet acquires an original channel following



up\*/down\* (Left-Right) path, it is not allowed to do transition to a new channel anymore to avoid deadlock situation.

#### IV. DESIGN METHODOLOGIES FOR ENERGY EFFICIENT NOC GENERATION

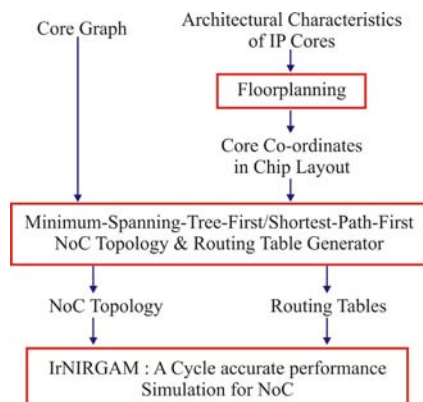


Fig.2 : Network construction using GA based

Based on the routing scheme presented by Silla et. al. (Silla & Duato, 2000), two novel genetic algorithm based methodologies referred as MSTF (minimum-spanning-tree-first) & SPF (shortest-paths-first) for energy efficient NoC topology generation are presented in this section. The presented methodologies generate an energy efficient customized NoC topology along with the required routing tables to provide deadlock free communication according to the communication requirement of the application under consideration. In both the presented methodologies, information from the floorplan and Core Graph exhibiting the chip layout and traffic characteristics respectively are taken as inputs as exhibited in Figure 2.

Assuming over the cell routing (Srinivasan & Chatha, 2006), the link length among the nodes in the chip layout can be taken according to Manhattan distance. In both the proposed methodologies, the link/channel length is not allowed to exceed the maximum permitted channel length ( $e_{max}$ ) due to constraint of physical signaling delay. This also prevents the algorithm from inserting wires that span long distances across the chip. Also, the nodes of the generated topology are not allowed to exceed a given maximum permitted node-degree ( $nd_{max}$ ). This constraint prevents the algorithm from instantiating slow routers with a large number of I/O-channels that would otherwise decrease the achievable clock frequency due to internal routing and scheduling delay of the router.

##### a) Minimum Spanning Tree First (MSTF) Methodology

In this proposed methodology to generate the energy efficient customized topology, first a minimum spanning tree (MST) using Prim's algorithm (Cormen, Leiserson & Rivest, 1990) is generated on the nodes of

the Core Graph according to information regarding the Manhattan distance from the floorplan with the constraints on  $nd_{max}$  and  $e_{max}$ . The node/core with maximum bandwidth requirement is assumed as the root of the tree. The minimum spanning tree in the topology helps us in classifying all the channels/links of the topology as "up" ("Left") or "down" ("Right"). The following phases of MSTF methodology helps in extending the network/topology for energy efficient deadlock free communication.

##### Energy Aware Topology Extension Phase :

While keeping the constraints on  $nd_{max}$  and  $e_{max}$ , the topology is further extended by laying the shortest energy path for each traffic characteristics (edges corresponding to pair of nodes in the Core Graph). Due to constraints on  $nd_{max}$  and  $e_{max}$ , the order in which such shortest energy paths are generated basically decides the total communication energy requirement of the generated topology. The optimized order of traffic characteristics of the application is found using a genetic algorithm (refer next section). The routing tables of nodes/routers in the discovered shortest energy path are updated with the routing table entry type tag as shortest path.

##### Deadlock Avoidance Phase :

Lastly the proposed methodology uses the modified Dijkstra's algorithm (Cormen, Leiserson & Rivest, 1990) according to up\*/down\* (Left Right) rule for finding deadlock free escape routing paths from each node in the shortest energy path to the corresponding destination in the generated NoC and tags them as up\*/down\* (Left-Right).

While taking routing decision the output channels tagged as shortest path are selected with higher priority and up\*/down\* (Left Right) tagged channels are selected only when no output channel corresponding to shortest path is free.

##### b) Shortest Path First (SPF) Methodology

SPF is similar to MSTF methodology with the exception that in SPF the topology generation is initiated by first finding the shortest energy path and later the topology is extended by constructing the MST. As in Energy Aware Topology Extension Phase of MSTF, a genetic algorithm is used to find the optimized energy-efficient traffic characteristics order of the application. Since in MSTF, MST is constructed first, it is possible that a large number of links for a number of nodes/cores in the topology are the links pertaining to MST. As maximum links emanating from a node is limited to  $nd_{max}$ , this phenomenon can lead to increased value of hop count in the shortest energy paths generated later leading to increased communication energy. However the SPF overcomes this drawback by creating the links pertaining to shortest energy path before the links pertaining to MST. As shortest energy paths in the topology are generated first in SPF and so there can be

a possibility that not enough number of free ports is available to construct the MST in the topology later. In such case a minimum number of ports per node/core need to be reserved before finding the shortest energy paths. However experiments showed that if communication requirement are uniformly distributed over the Core Graph then such problems are rare if any. Algorithm 1 briefly presents the proposed methodologies.

#### Algorithm 1 : MSTF & SPF Design Methodology

##### Require :

1.  $\mathcal{C} = \text{CG} = \text{Core Graph} = \{E \text{ edges (i.e. traffic characteristics)}, V \text{ vertices}\}$
2.  $V = \{v_i \mid v_i \text{ is } i^{\text{th}} \text{ IP core}\}$
3.  $E = \{e_{ij} : v_i \rightarrow v_j \text{ with weight } bw_{ij} \mid v_i \text{ (source)}, v_j \text{ (destination)} \in V\}$
4.  $\text{NoC} = \{T \text{ (Topology)}, R \text{ (Set of routing tables)}, S \text{ (set of shortest path)}\}$
5.  $\text{TC\_Array} = \{\text{Array of traffic characteristic (i.e. ordered set of } E)\}$
6.  $nd_{\max}$  = Maximum permitted node degree in the topology  $T$
7.  $e_{\max}$  = The maximum permitted length of a link(channel) in topology  $T$
8. Manhattan Distance =  $\Delta = \{d_{ij} \mid d_{ij} = |v_i - v_j|, v_i, v_j \in V\}$
9. Manhattan Distance greater than  $e_{\max}$  are not considered.

**Ensure :** Energy Aware NoC Topology for CG

##### Procedure Minimum-Spanning-Tree-First ()

```

x • .NoCEA ;
// initialize the energy aware NoC (i.e. NoCEA)
• NoCEA.T =  $\Phi$ ; NoCEA.R =  $\Phi$ ; NoCEA.S =  $\Phi$ ;
•  $\Gamma = \{\text{minimum spanning tree as per } \Delta \text{ with constraint } nd_{\max} \text{ \& } e_{\max}, \text{ root is node with maximum communication in } \mathcal{C}\}$ 
• NoCEA.T = NoCEA.T  $\cup \{\Gamma\}$ 
• (NoCEA, TC_Array) = GeniticAlgo(NoCEA,  $\Gamma$ )
• for each path  $s_i \in S$  in NoCEA.S
  o N = {set of nodes in path  $s_i$ }
  o for  $n_j \in N$ 
    ▪ NoCEA.R = NoCEA.R  $\cup \{\text{update routing tables in NoC}_{EA}.R \text{ for nodes } \in V \text{ in the root followed by the shortest up*/down* (Left-Right) escape path from node } n_j \text{ to the destination node of path } s_i. \text{ The routing Table entry type tag is set as up*/down* (Lef -Right) for these nodes}\}$ 
  o endfor
• endfor

```

##### Endprocedure

##### Procedure Shortest-Paths-First ()

```

x • .NoCEA ;
// initialize the energy aware NoC (i.e. NoCEA)
• NoCEA.T =  $\Phi$ ; NoCEA.R =  $\Phi$ ; NoCEA.S =  $\Phi$ ;
•  $\Gamma = \Phi$ ;
• (NoCEA, TC_Array) = GeniticAlgo(NoCEA,  $\Gamma$ )
•  $\Gamma = \{\text{minimum spanning tree as per } \Delta \text{ with constraint } nd_{\max} \text{ \& } e_{\max}, \text{ root is node with maximum communication in } \mathcal{C}\}$ 

```

```

• NoCEA.T = NoCEA.T  $\cup \{\Gamma\}$ 
• for each path  $s_i \in S$  in NoCEA.S
  o N = {set of nodes in path  $s_i$ }
  o for  $n_j \in N$ 
    ▪ NoCEA.R = NoCEA.R  $\cup \{\text{update routing tables in NoC}_{EA}.R \text{ for nodes } \in V \text{ in the root followed by the shortest up*/down* (Left-Right) escape path from node } n_j \text{ to the destination node of path } s_i. \text{ The routing table entry type tag is set as up*/down* (Lef -Right) for these nodes}\}$ 
  o endfor
• endfor
endprocedure

```

## VII. GENETIC ALGORITHM

A genetic algorithm (Eiben & Smith, 2003) based heuristic is used to find the best order of the traffic characteristics to generate the shortest energy paths in topology such that the communication energy requirement of the application is optimized. Genetic algorithm is a search technique used in determining exact or approximate solutions to optimization and search problems. Genetic algorithms are a particular class of evolutionary algorithms which uses techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover. The proposed genetic algorithm explores the search space extensively to generate an irregular topology with optimized communication energy requirement for the given application. The proposed genetic algorithm formulation is as follows.

### a) Solution Space

In formulation of the proposed methodology, each chromosome is represented as an array of genes. Maximum size of the gene array is equal to the number of edges in the Core Graph. Each gene of the chromosome represents a traffic characteristic (an edge corresponding to a pair of nodes in the Core Graph)

### b) Initial Population

A large population (i.e. 500 chromosomes) of chromosome is initially generated. The chromosomes of the initial population are generated by assigning traffic characteristics of the application to the chromosome's gene array in some random order. The initial population is later sorted according to the increasing order of total communication energy requirement of the generated topology (chromosome). It is worth highlighting here that the communication energy consumption by a chromosome varies depending on the traffic characteristics order (order of elements in gene array) of the chromosome.

### c) Crossover

In each generation, crossover is performed on 50% of the population with the bias towards the Best Class of the chromosome population. For achieving crossover of two chromosomes, a random crossover point is selected. Two new chromosomes are created by

the crossover operation. The new chromosomes are created by copying the traffic characteristics (genes) from their respective parents till crossover point or from crossover point to the end of the chromosome and then the remaining traffic characteristics (genes) are copied according to the order of traffic characteristics (genes) in the other chromosome such that there are no duplicate traffic characteristics in the created chromosomes.

#### d) Mutation

In each generation, mutation is performed on 40% of the population to avoid the solution from getting stuck up in the local minima. Two types of mutations with probability of 50% each are performed in each generation. In first type of mutation a gene in the gene array of the chromosome with highest energy requirement is swapped with a randomly selected gene of the chromosome. In second type two randomly selected genes in the gene array of the chromosome are swapped.

#### e) Measure of Fitness

The cost function used to measure the fitness of the chromosomes in the population can be formulated as under.

$$\text{Cost} = E_{c_i} / X$$

Where X is maximum chromosome energy requirement among all the chromosomes in the population,  $E_{c_i}$  is the energy requirement for chromosome  $c_i$ . Fitness of chromosome is regarded as high if its cost approaches 0. It may be noted that, the best 10% chromosomes (referred as Best Class) in any generation are directly transferred to the next generation so as not to degrade the solution between the generations.

Algorithm 2 briefly presents the proposed genetic algorithm formulation. After genetic algorithm methodology is made to run for a required number of generations, the NoC topology and routing tables corresponding to the best output chromosome are accepted as the customized energy optimized application specific NoC.

*Algorithm 2*: Genetic Algorithm (GA) formulation of energy aware application specific NoC generator

**procedure** GeniticAlgo(  $\kappa$  NoCEA, T  $\Gamma$  )

- $\mu$  = % of chromosomes for mutation
- $\xi$  = % of chromosomes for crossover
- $\lambda$  = % of chromosomes retained in next generation
- $G = \{\text{gene array}[] \mid \text{size}(\text{gene array}[]) = |E| \text{ (i.e. | traffic characteristics |)}\}$
- $C = \text{chromosome} = \{G \text{ (set of genes), } \kappa \text{ (corresponding NoC)}\}$
- Chromosome Population = CSet =  $\{C_i \mid C_i \text{ is } i^{\text{th}} \text{ chromosome with gene array } G_i \text{ and associated NoC } \kappa_i\}$

- CSet CSet = Generate\_Initial\_Population(NoCEA, CSet)
- **while** (number of generations not attained)
  - Sort CSet in ascending order of cost ( i.e. total communication energy)
  - Keep first  $\lambda$  fraction chromosomes of CSet for next generation as Best Class
  - Generate next  $\xi$  fraction chromosomes for next generation with crossover operations on CSet
  - Select a random pair ( $C_1, C_2$ ) of chromosomes from CSet with bias towards Best Class
  - ( $C'_1, C'_2$ ) = CrossOver( $C_1, C_2, \Gamma$ )
  - Generate the remaining  $\mu$  fraction of chromosomes for next generation with mutation operations on CSet
  - Randomly Select a chromosome  $C_i$  from CSet
  - Select random  $r \in \{1, 2\}$
  - Mutation( $C_i, r, \Gamma$ )
- **endwhile**
- Sort CSet in ascending order of cost (i.e. total communication energy)
- $C_{\text{best}} = C_{\text{Set}}[0]$
- **return** ( $\kappa_{\text{best}}$  (NoC),  $G_{\text{best}}$  (Gene Array) corresponding to  $C_{\text{best}}$ )
- endprocedure**

## VIII. EXPERIMENTAL RESULTS

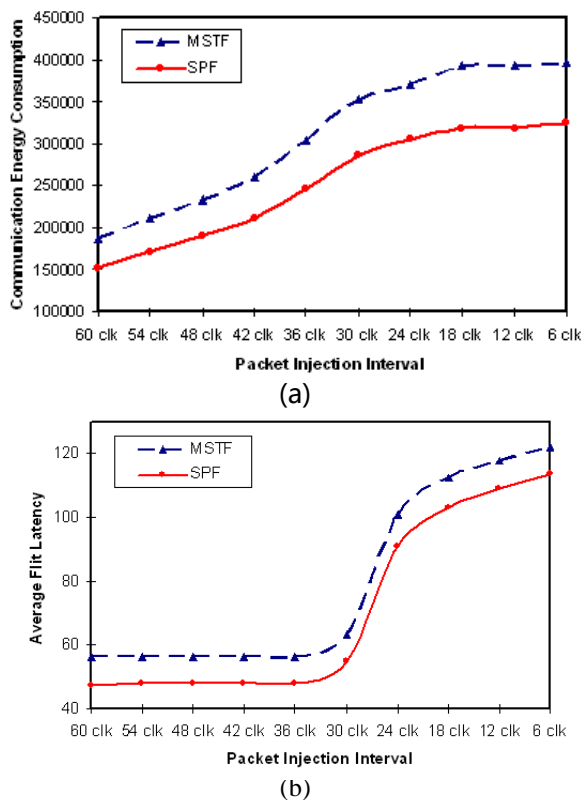
The generated energy aware application specific topology was evaluated with respect to the communication energy consumption with applied traffic load on the NoC simulation framework. In order to obtain a broad range of different irregular traffic scenarios, multiple Core Graphs using TGFF (Dick, Rhodes & Wolf, 1998) were randomly generated with diverse bandwidth requirement of the IP Cores. For performance comparison, a NoC simulator IrNIRGAM, the extended version of NIRGAM (Jain, Al-Hashimi, Gaur, Laxmi & Narayanan, 2007; Jain 2007) supporting irregular topology with the provision of supporting escape path routing for avoiding deadlock condition, was deployed. IrNIRGAM is a discrete event, cycle accurate simulator. IrNIRGAM supports irregular topology framework with source and table based routing in a wormhole switching based architecture wherein an IP Core is directly connected to a dedicated router. In IrNIRGAM, input buffered routers can have multiple virtual channels (VCs) and uses wormhole switching for flow control. The packets are split into an arbitrary number of flits (flow control units) and forwarded through the network in a pipelined fashion. A Round-Robin scheme for switch arbitration is used in the router nodes to provide fair bandwidth allocation while effectively preventing scheduling anomalies like starvation. For performance comparison on experimental set, the IrNIRGAM was run for 10000 clock cycles with applied packet injection interval to evaluate the network performance with varying traffic load. The energy consumption by the flits reaching their corresponding destination and flit latency were used as performance metric. The energy consumption by router

in transmitting a bit is evaluated using the power simulator orion (Kahng, Li, Peh & Samadi, 2009) for  $0.18\mu\text{m}$  technology. Similarly the dynamic bit energy consumption for inter-node links (Elbit) can be calculated using the following equation.

$$El_{bit} = (1/2) \times \alpha \times C_{phy} \times V_{DD}^2$$

Where  $\alpha$  is the average probability of a 1 to 0 or 0 to 1 transition between two successive samples in the stream for a specific bit. The value of  $\alpha$  can be taken as 0.5 assuming data stream to be purely random.  $C_{phy}$  is the physical capacitance of inter-node wire under consideration for the given technology and VDD is the supply voltage.

a) *Experiments on SPF and MSTF with Random Benchmarks*

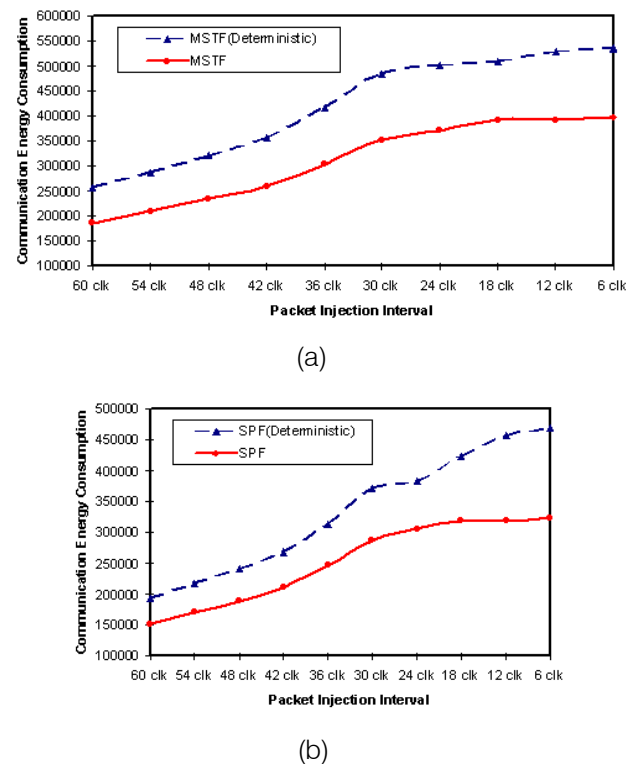


**Fig.3 :** Performance comparison with varying packet injection interval of (a) communication energy consumption (in pico joules) and (b) Average flit latency (in clock cycles) of the proposed Minimum-Spanning-Tree-First (MSTF) and Shortest-Path-First methodologies (SPF).

The performance of the proposed Shortest-Path-First Methodology (SPF) and Minimum-Spanning-Tree-First Methodology (MSTF) were compared on the IrNIRGAM simulation framework with varying packet injection interval (i.e. varying communication traffic load). Figure 3 shows performance results averaged over 50 generated energy efficient irregular topologies generated based on up\*/down\* routing function with varying number of cores from 16 to 81, ndmax = 4 and

permitted channel length (emax) was taken as 1.5 times the length of the core/node with largest length among all the cores in the NoC. The proposed shortest-path-First (SPF) methodology's total dynamic communication energy consumption was on average 18.5% lesser in comparison to minimum-spanning-tree-first (MSTF) methodology in addition to reduced latency ( in the range of 7.5 clocks to 10 clocks) for equivalent throughput.

b) *Experiments on SPF/MSTF and SPF (Deterministic)/MSTF (Deterministic) with Random Benchmarks*



**Fig.4 :** Performance comparison with varying packet injection interval of dynamic communication energy consumption (in pico joules) of the (a) MSTF and MSTF (Deterministic) and (b) SPF and SPF (Deterministic).

The performance of the proposed Genetic algorithm based Shortest-Path-First Methodology (SPF) and Minimum-Spanning-Tree-First Methodology (MSTF) were compared with deterministic methodologies MSTF (Deterministic) and SPF (Deterministic) proposed in (Choudhary, N., Gaur, M. S., Laxmi, V., Singh, V., 2010) of order  $O(n^2)$ . IrNIRGAM simulation framework was run for 10000 clock cycles with varying packet injection interval (i.e. varying communication traffic load) . Figure 4 shows comparison of the dynamic communication energy consumption by the proposed methodologies and the work proposed in (Choudhary, N., Gaur, M. S., Laxmi, V., Singh, V., 2010). The experimental results were averaged over 50 generated customized irregular topologies generated based on up\*/down\* routing function with varying number of cores from 16 to 81,



$nd_{max} = 4$  and permitted channel length ( $e_{max}$ ) was assumed as 1.5 times the length of the core/node with largest length among all the cores in the NoC. The proposed MSTF methodology's total dynamic communication energy consumption was on average 26.2% lesser in comparison to MSTF (Deterministic) whereas for the proposed SPF the total dynamic communication energy consumption was on average 24.3% lesser in comparison to SPF (Deterministic) for equivalent throughput.

### c) Experiments on SPF, MSTF and Regular NoC with Random Benchmarks

To compare the performance of the proposed methodologies with regular NoC, the performance of the proposed methodologies with up\*/down\* and Left-Right routing function were compared with 2D-Mesh NoC with XY and OE routing for the packet injection intervals according to the application's traffic characteristics. The sizes of the tiles are kept same in the proposed methodologies as in regular 2D-Mesh. Figure 5 shows the performance comparison of MSTF with 2D-Mesh averaged over 50 generated energy efficient irregular topologies with varying number of cores from 16 to 81,  $nd_{max} = 4$  and  $e_{max}$  was taken as 2 times the length of the core/node. The MSTF with up\*/down\* (Left-Right) routing shows reduced average flit latency in the range of 5.8 (4.4) clocks to 13.3 (15.2) clocks and 9.6 (8.2) clocks to 68 (67) clocks in comparison to 2D-Mesh with XY and OE routing respectively. The average per flit communication energy comparison of MSTF with 2D-Mesh shows reduction in the range of 10% (8%) to 21% (19%) and 18% (17%) to 46% (46%) in comparison to XY and OE routing respectively for up\*/down\* (Left-Right) routing.

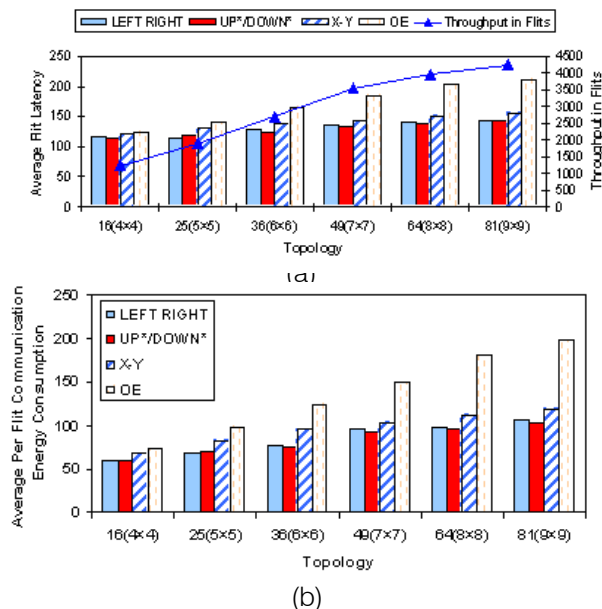


Fig.5 : MSTF performance comparison with 2D-Mesh (a) Average flit latency (in clock cycles) and (b) Average communication energy consumption per flit (in pico joules)

The average per flit communication energy comparison of SPF with 2D-Mesh shows reduction in the range of 18.8% (18.5%) to 29.2% (25.8%) and 25.2% (24.6%) to 54.7% (53%) in comparison to XY and OE routing respectively for up\*/down\* (Left-Right) routing.

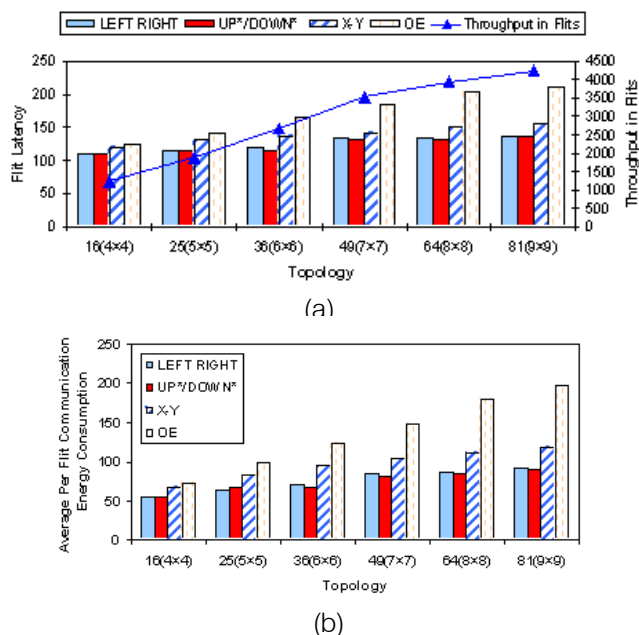


Fig.6 : SPF performance comparison with 2D-Mesh (a) Average flit latency (in clock cycles) and (b) Average communication energy consumption per flit (in pico joules).

The above mentioned results shows that the performance of Left-Right and up\*/down\* routing function for MSTF and SPF depends on the traffic characteristics and the corresponding generated topology i.e. one routing function performs better than other depending on the traffic characteristic and the corresponding generated topology. However we have observed that up\*/down\* routing tends to perform better in most of the cases. Moreover the performance comparison between MSTF and SPF clearly shows that in most cases the SPF methodology performs reasonably better than MSTF methodology.

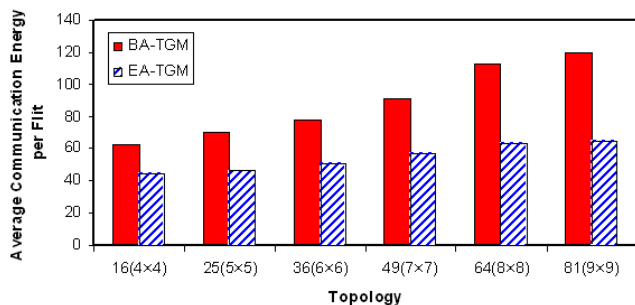
Figure 6 shows the SPF performance results. The SPF with up\*/down\* (Left-Right) routing shows reduced average flit latency in the range of 10 (9.4) clocks to 20.9 (18.4) clocks and 13.8 (13.2) clocks to 76 (69) clocks in comparison to 2D-Mesh with XY and OE routing respectively.

### D. Experiments on SPF, 2D-Mesh, and BA-TGM

The per flit dynamic communication energy consumption for proposed SPF and Bandwidth Aware Topology Generation Methodology (referred as BA-TGM) presented in (Choudhary, N. et al., 2010) are compared for 50 generated customized irregular topologies with cores having varying sizes and  $nd_{max}$  of 4 for number of cores varying between 16 to 81. For BA-TGM, up\*/down\* routing was assumed whereas for SPF escape path based up\*/down\* routing was used. The

emax was taken as 1.5 times the length of the core having maximum length among all the cores of the NoC.

Figure 7 shows that SPF consistently performs better in comparison to BA-TGM as far as average dynamic communication energy consumption by flits reaching their destination is concerned. The SPF showed on average a reduction of 38.6% for the communication energy per flit in comparison to BA-TGM.



*Fig. 7:* Comparison of average communication energy consumed by flits in reaching their destination for BA-TGM and SPF with ndmax = 4

## IX. CONCLUSION

In this paper, the energy efficient customized Irregular topology generation problem for NoC was addressed. Two genetic algorithm based novel methodologies are proposed for generating the NoC topology with optimized communication energy requirements according to the traffic characteristics of the given application. Although in this paper up\*/down\* and Left-Right routing were used as escape path for deadlock prevention, we argue that the proposed methodologies can be adapted with any of the topology agnostic routing algorithms where generic routing rules based on turn prohibition can be enforced. It is believed that the combined treatment of the routing and topology generation as done in the presented methods offers a huge potential of optimization for future application-specific NoC architectures.

Some interesting extensions of the proposed design can be to combine the topology generation with the task partitioning/scheduling into the presented framework to make the design more adaptable to the dynamic communication requirement of the application in such a way that the computation and communication energy consumption can be optimized at the same time.

## REFERENCES REFERENCES REFERENCIAS

1. Ahonen, T. et al. (2004), Topology optimization for application specific networks on chip. In Proceedings SLIP.
2. Benini, L., & DeMicheli, G. (2002). Networks on Chips: a new SoC paradigm. In IEEE Comput. 35, 70–78.
3. Benini, L., & DeMicheli, G. (2002). Networks on Chips: a new SoC paradigm. In IEEE Comput. 35, 70–78.
4. Chang, Y. C., Chang, Y. W., Wu, G. M. Wu, S. W. (2000). B\*-Trees : a new representation for non-slicing floorplans. In Proceeding of 37th Design Automation Conference, 458-463.
5. Cormen, T., Leiserson, C. & Rivest, R. (1990). Introduction to algorithms, Prentice Hall International.
6. Dally, W. J., & Towles, B. (2001). Route packets, not wires: on-chip interconnection networks. In IEEE Proceedings of the 38th Design Automation Conference (DAC), 684-689.
7. Dally, W., & Seitz, C. (1987). Deadlock-free message routing in multiprocessor interconnection networks. In IEEE Transactions on Computers, 547–553.
8. Dick, R. P., Rhodes, D. L., & Wolf, W. (1998). TGFF: task graphs for free. In Proceeding of the International Workshop on Hardware/Software Codesign.
9. Duato, J., Yalamanchili, S. & Ni, L. (2003). Interconnection networks: an engineering approach, Elsevier.
10. Eiben, A. E., & Smith, J. E. (2003). Introduction to evolutionary computing, Berlin, Heidelberg. Springer-Verlag.
11. Glass, C. & Ni, L. (1992). The turn model for adaptive routing. In Proceeding of 19-th International Symposium on Computer Architecture, 278–287.
12. Hansson, A. et al. (2005). A unified approach to constrained mapping and routing on network-on-chip architectures. In Proceeding of ISSS, 75-80.
13. Ho, W. H., & Pinkston, T. M. (2003). A methodology for designing efficient on-chip interconnects on well-behaved communication patterns. In HPCA, 377-388.
14. Hu, J. & Marculescu, R. (2003). energy-aware mapping for tile-based NoC architectures under performance constraints. In ASP-DAC.
15. Hu, J., & Marculescu, R. (2005). Energy- and performance-aware mapping for regular NoC architectures. In IEEE Trans. on CAD of Integrated Circuits and Systems, 24(4).
16. Jain, L., (2007) Network on Chip simulator: NIRGAM. Retrieved October 17, 2010, from <http://www.nirgam.ecs.soton.ac.uk>
17. Jain, L., Al-Hashimi, B. M., Gaur, M. S., Laxmi, V., & Narayanan, A. (2007). NIRGAM: a simulator for NoC interconnect routing and application modelling. In proceedings of DATE.
18. Jouraku, A., Funahashi, A., Amano, H., & Koibuchi, M. (2001). L-turn routing: an adaptive routing in

- irregular networks. In Proceeding of the International Conference on Parallel Processing, 374-383.
19. Kahng, A. B., Li, B. L., Peh, S., & Samadi, K. (2009). Orion 2.0: a fast and accurate NoC power and area model for early-stage design space exploration. In Proceedings DATE, 423-428.
  20. Kumar, S., Jantsch, A., Soininen, J. P., Forsell, M., Millberg, M., Oberg, J., Tiensyrja, K., & Hemani, A (2002). A network on chip architecture and design methodology. In Proceedings of VLSI Annual Symposium (ISVLSI 2002), 105-112.
  21. Lahiri, K. et al. (2004). Design space exploration for optimizing on-chip communication architectures. In IEEE TCAD, 23(6), 952- 961.
  22. Lin, J. M. & Chang, Y. W. (2005). TCG : A transitive closure graph-based representation of general floorplans. In IEEE Transactions on VLSI Systems, 288-292.
  23. Murali, S. & De Micheli, G. (2005). An application-specific design methodology for STbus crossbar generation. In Proceedings DATE. 1176-1181.
  24. Murali, S. et al. (2005). Mapping and physical planning of networks on chip architectures with quality-of-service guarantees. In Proceedings ASPDAC.
  25. Murali, S., & DeMicheli, G. (2004). SUNMAP: a tool for automatic topology selection and generation for NoCs. In Proceeding of DAC.
  26. Murali, S., Seiculescu, C., Benini, L., & De Micheli, G. (2009). Synthesis of networks on chips for 3d systems on chips. In Asian and South Pacific Design Automation Conference (ASPDAC), 242-247.
  27. Natvig, L. (1997). High-level architectural simulation of the torus routing chip. In Proceedings of the International Verilog HDL Conference, California, 48-55.
  28. Choudhary, N., Gaur, M. S., Laxmi, V., Singh, V. (2010). Fast Energy Aware Application Specific Network-on-Chip Topology Generator. In Proceeding of the IEEE International Conference IACC, Patiala, India, 250-255
  29. Choudhary, N. et al. (2010). Genetic Algorithm Based Topology Generation for Application Specific Network-on-Chip. In Proceeding of the IEEE International Conference ISCAS, Paris, France, 3156-3159
  30. Ogras , U., Hu, J., & Marculescu, R. (2005). Key research problems in NoC design: a holistic perspective. In IEEE CODES+ISSS, 69-74.
  31. Pinto A. et al. (2003). Efficient Synthesis of Networks on Chip. In ICCD, 46-150.
  32. Ravi, R. et al. (2001). Approximation algorithms for degree-constrained minimum cost network design problems. In Algorithmica, 31(1), 58-78.
  33. Schroeder, M. D. et al., (1991). Autonet: a high-speed self-configuring local area network using point-to-point links. In Journal on Selected Areas in Communications, 1(9).
  34. Seiculescu, C., Murali, S., Benini, L., & De Micheli, G. (2009). SunFloor 3D: a tool for networks on chip topology synthesis for 3d systems on chip. In Proceedings DATE, 9-14.
  35. Silla, F. et al. (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, 46-60.
  36. Silla, F., & Duato, J. (2000). High-performance routing in networks of workstations with irregular topology. In IEEE Transactions on Parallel and Distributed Systems, 11(11), 699-719.
  37. Srinivasan, K. & Chatha, K. S. (2006). Layout aware design of mesh based NoC architectures. In Proceedings of 4th International Conference on Hardware Software Codesign and System Synthesis. Seoul, Korea, 36-141.
  38. Srinivasan, K. et al. (2005). An automated technique for topology and route generation of application specific on-chip interconnection networks. In Proceedings ICCAD.
  39. Srinivasan, K. & Chatha, K. S. (2005). ISIS: A genetic algorithm based technique for custom on-chip interconnection network synthesis. In Proceedings of 18th International Conference on VLSI Design, Kolkata, India, 623-628.
  40. Sun, Y. M., Yang, C. H., Chung, Y. C., & Hang, T. Y. (2004). An efficient deadlock-free tree-based routing algorithm for irregular wormhole-routed networks based on turn model. In Proceeding of International Conference on Parallel Processing, 1(1), 343-352.
  41. Wu, J. & Sheng, L. (1999). Deadlock-free routing in irregular networks using prefix routing. DIMACS (Tech. Rep.), 99-19.