Analyzing Methodologies of Irregular NoC Topology Synthesis

Naveen Choudhary
Department of CSE
College of Technology and Engineering, MPUAT
Udaipur, India

Dharm Singh
Department of CSE
College of Technology and Engineering, MPUAT
Udaipur, India

Surbhi Jain
Department of CSE
College of Technology and Engineering, MPUAT
Udaipur, India

ABSTRACT
Network-On-Chip (NoC) provides a structured way of realizing communication for System on Chip (SoC) with many processing cores, which emphasize a communication-centric, as opposed to a computation-centric, design view. Network-on-Chip architectures have a wide variety of parameters that can be optimized according to the designer's requirements. Exploration and optimization of these parameters is an active area of research and a large number of methodologies have been proposed for this. In this paper we study the existing techniques and categorize them on the basis of considered optimization objectives.

Keywords
Network-on-Chip (NoC), NoC communication graph, Genetic algorithm, Particle swarm optimization, Ant colony optimization.

1. INTRODUCTION
The challenge of communication aspect in SoCs has attracted attention of many researchers. A promising solution that has come up from the researches in this field is NoC (Network on Chip). The concept of NoC [1-3] can be viewed as a unification of on-chip communication solutions rather than an explicit new alternative. The on-chip interconnection networks (i.e. communication infrastructure) represent one of the major elements which have to be optimized in designing complex ultra large scale SoC systems. The International Technology Roadmap for Semiconductors [4] foresees that it will face the limiting factor of performance and power consumption in next generation ultra large scale systems-on-chip (SOCs).

One of the limiting factors in enhanced NoC communication performance is the topological design of the interconnection between the various PE(Processing Elements). This is due to the fact that NoC-based interconnection performance correlates strongly with the topology selected for implementation. Therefore selection of an appropriate topology plays a vital role in the synthesis process. Early research emphasized the use of regular topologies like tori, mesh etc. The driving factor behind it was the simplicity of the topology structure. However irregular network architectures might be necessary for realizing application-specific SoCs [5], such as those in mobile-phone systems, where different heterogeneous blocks with varying communicating requirements must be linked. When implementing standard topologies which are fault tolerant irregularity needs to be taken into account, irregular mesh topology being an example. While floor planning process clustering is required for cores which need to communicate frequently and so need to be placed next to each other. Standard topologies may not cater to such a requirement efficiently.

Topology generation constitutes only a single phase in NoC synthesis [6]. The other phases are task identification, core to tile mapping, floorplanning and optimization. This order may not be same in all synthesis process. In this paper we study different optimization approaches which follow different order of NoC synthesis with multiple optimization objectives.

The rest of the paper is organized as follows. The next section Section II gives a brief of area optimization techniques which aim at reducing NoC floor plan area. Section III lists the methodologies which minimize energy or power consumption as the highest priority. The last section presents the summary and possibilities for future work.

2. AREA OPTIMIZED NoC SYNTHESIS
This section gives a brief description of the methodologies which give area optimization a higher priority while achieving other optimization objectives as well. Floor planning is either carried out using different technique or is a phase in proposed methodology.

2.1 ISIS: genetic algorithm based technique
The genetic algorithm (GA) based technique in [7], [8] focuses on application specific communication architecture synthesis and routing. It takes number of ports maximum bandwidth, latency and power consumption as router specification inputs.

Problem is defined in the form of communication trace graph where each vertex is a processing element/memory unit; edge is the communication trace between two vertices and the weight of the edge is determined by bandwidth requirement and latency constraints. With an objective to avoid congestion bandwidth constrains are not violated on any router port at any stage during NoC architecture synthesis. It is assumed that in the uncongested state power consumption in NoC is linearly proportional to traffic flowing through network. Thus to minimize power it proposes to minimize traffic flow. For area optimization total numbers of routers are minimized. To eliminate possibility of deadlock additional virtual channels are added after the NoC architecture has been synthesized.

The technique models the random topology population in hierarchical form where first level accounts for number of routers represented by I instances of binary array *arstr*, second represents CTG nodes-router port mapping stored in integer array *npstr* with J instances and third represents routing from sink to source denoted by linked list *trlist* of integer arrays *trstri*. Fitness is evaluated as a weighted function of power, area consumption and number of unmapped traces. At each level of hierarchy new generation of solution is produced by

applying genetic operators viz. mutation and crossover. For crossover equal sized arrays are cut at same location and exchanged. For mutation at the router level bit inversion is performed. To mutate the mapping two ports are selected at random and exchanged; for mutation at trace level the proposed methodology selects an unmapped trace and maps it to architecture using MSP algorithm. At each level solution produced by crossover outnumbers those produced by mutation.

The results are compared with MILP generated solutions. Benchmarks used are mp3 audio encoder, mp3 audio decoder, H.263 video encoder, and H.263 video decoder algorithms. The results shown that though the runtime of ISIS in higher for small graphs it does not grow factorially with the size as observed in MILP formulation [9].

2.2 Fat tree based optical NoC

In [10] a methodology is proposed for a new optical NoC based on fat-tree-network. It covers its protocols, topology, floor plan, and a low-power and low-cost optical router, optical turnaround router (OTAR).

Making use of non blocking property of OTAR which is an optical turn around router, turn around algorithm can be implemented for routing. This property of OTAR router is suggested to help in increasing network throughput. OTAR can passively route packets without powering on any microresonator in 40% of all cases

The suggested network can transmit both payload and control packets differentiated by type of switching used. It uses circuit switching for payload data and packet switching is considered for network control data.

It has a hierarchical network topology structure which is suggested to aid in building of a strong multiprocessor system while saving on are utilization as it can take form of 3-D NoC structure.

The proposed network is analyzed in terms of power consumption, optical power loss, and network performance. An analytical model is developed to assess the power consumption of fat tree-based optical NoC (FONoC). Based on the analytical model and SPICE (Simulation Program with Integrated Circuit Emphasis) simulations, FONoC is compared with a matched electronic NoC in 45nm technology. The results show that FONoC can save up to 87% power to achieve the same performance for a 64-core MPSoC. FONoC is simulated for a 64-core MPSoC and it show the end-to-end delay and network throughput under differed offered loads and packet sizes.

2.3 Genetic algorithm based congestion aware topology generation

In [11], [12] a GA based technique is used for topology generation under the assumption that mapping of tasks to IP core already been carried out. IrNIRGAM [5] an extension of NIRGAM [13] used for simulation and supports wormhole switching, source and table based routing and virtual channels.

Input to the methodology is a core graph where vertices represent an IP core and edge denote the communication link between the respective cores. Weight of edges accounts for desired average bandwidth. The connectivity and available link bandwidth is represented through NoC topology graph.

Energy model is referenced from [14]. Round robin is used for switch arbitration. Routing functions used are up*/down* and L-R routing. These functions are distributed in nature and are implemented using distributed table based routing.

Topology generation starts with first step as floor planning. With core graph as input minimum spanning tree created using Prim's algorithm based on *Manhattan distance* between IP cores. Next GA applied to generate customized topology. Chromosome denotes NoC topology and genes are assumed as collection of deadlock free path. For initial population generation shortest deadlock free path are generated applying Dijkstra's shortest path algorithm. Routing table entries are made for each edge in core graph. Traffic load assigned as per bandwidth requirement. Mutation is carried out in accordance with up*/down* rule and permitted node degree and permitted channel length constraints are not violated to ensure that link/channel length does not exceed the maximum permitted limit and slow channels avoided.

Three mutation operators are used in the proposed methodology with equal probability. First one added shortcut path to reduce load on heavily loaded paths. The second step removes lightly loaded paths and distributes the traffic in existing paths. The third step reduces the energy consumption by finding shorter replacement for each path of every gene. Crossover is performed with bias towards the best class population while not violating maximum permitted node degree and link length constraint.

Proposed algorithm was termed BA-TGM. Results are compared with 2D-mesh with same tile size and task to core/tile mapping [14]. Better performance is observed with respect to average communication energy by the proposed methodology. Applications used for experiment were random benchmarks as well as intelligent Task to Core mapping.

3. INTERCONNECTION OPTIMIZED NoC SYNTHESIS

This section covers the methodologies in which the optimization objectives do not include area or in other words floor planning is not emphasized. These mainly concentrate on optimizing communication energy or cost of communication.

3.1 Ant colony optimization

In [15] it is argued that larger link width results in increasing buffer size and also power consumption, moreover the increase in link operational frequency also increases the overall power consumption of the system. Therefore optimizing bandwidth should account for both the above factors. The mapping of an application to the NoC platform which is an important phase in NoC synthesis as it affects the performance of the system in this methodology 2D-mesh is used as the underlying topology. Wormhole routing technique is implemented along with deterministic XY routing algorithm. Router with input queues and multiple virtual channels were assumed.

A network topology representation is similar to the ones used in [7-8], [10-12] with vertices representing the processors and weight of the links represents its bandwidth. Application task graph, network topology graph and routing function are used as input to the proposed methodology. The total bandwidth requirement *B* signifies the sum of bandwidth required of all links' bandwidths requirement post the mapping phase.

To apply the Ant Colony Optimization (ACO) values of number of iterations, ants and probability between processors and task mappings are initialized. Each ant carries out task-processor mapping at random with certain probability. At the

end of each iteration the ant which obtains the best optimized result can update the pheromone i.e. the probability. After several iterations a near optimal solution could be generated. Total bandwidth is used as a parameter for fitness evaluation.

Results of the execution are compared with realistic benchmarks such as MPEG 4 decoder and VOPD applications.

3.2 Particle swarm optimization

In [16] a particle swarm optimization (PSO) based topology independent NoC design methodology is presented. Core graph and NoC architecture graph are taken as input. Reducing hop count can decrease fault tolerant capacity of NoC therefore the approach emphasize on minimizing communication cost while maximizing fault tolerance. In a PSO approach, multiple candidate solutions called particles wander in problem space. The path is dependent on experience of its own as well its neighbor's. It utilizes two information indexes which are velocity and position for searching the problem space. The next position or direction is calculated as a function of current velocity and position vector.

The authors consider *onyx* as one of the best mapping algorithm and utilize it to generate initial population of particles. To avoid rapid convergence in the technique applied the velocity threshold is not defined and constants are assigned initial values obtained by observing several simulations. Due to convergence control different mappings are generated for the given application which covers a wide range of communication cost.

The algorithm used is Application-Specific Channel Dependency Graphs (ASCDG) which provides deadlock freedom and is highly adaptive and therefore assumed to be suitable for application specific NoC. In this paper path-based contention is considered as a mapping factor. After different mappings are obtained, robustness index and contention factor are used to select the most optimized application as per the requirement of designer. For evaluation, a total cost function is used which is a combination of cost of communication, robustness and contention factor. For experimental purpose the algorithm is applied to VOPD and MPEG-4 applications and achieves reasonably good results.

3.3 Hierarchical genetic algorithm

The algorithm proposed in [17] is a combination of genetic algorithm and swarm intelligence (PSO) approach. Traditional genetic algorithms are argued to have limitation like if crossover operation generates solutions different from original population then determining the IP it represents is difficult and if the solution is identical to that produced in initial population then the mathematical model may not hold true. Therefore a hierarchical genetic algorithm is suggested in this paper. Communication power model is same as used in Hu [14]. Fitness function is the energy model which takes into account Manhattan distance and communication between nodes. Three transformation methods are applied to produce new generation. One of them selects two adjacent elements, other selects two elements randomly and the third selects a 2 × 2 sub-matrix randomly and transform it in clockwise order. The task graph is taken as input and task matrix is generated by calculating the weight of each communication node in decreasing order. The selected first four nodes which is 4 × 4 matrix is divided into two layers. The most inner 4 cells are the first layer and the outer 12 cells as the second layer. For initialization, 4 IP cores with heavy communication traffic are randomly mapped to inner layer and the others to outer layer as this helps in rapid convergence. This results in initialization of 30 individuals. During iterations fitness of every individual is calculated and 10 minimal solutions are carried to the new population. Individuals are then selected randomly from the 10 minimal solutions and three transformation methods applied to generate 8, 8 and 4 children respectively at the end of each iteration. The process is carried out for fixed number of iterations. The experiments comparison of [16] with random IP cores mapping method shows up to 39% saving in communication energy.

3.4 Multi objective evolutionary algorithm

The algorithm suggested in [18] works in 2 phases. Phase1 computational synthesis generates core communication graph and second phase is communication synthesis the output of it is energy and throughput synthesized NoC backbone structure. Considering a 2-D Mesh communication architecture where each switch has a small buffer registers to avoid packet loss. Static XY wormhole routing is used for deadlock free communication. For energy calculation the approach adds computation energy consumption to the energy model used in Hu [14]. It uses "*Pareto archive*" (elitism) concept where best solutions encountered over the generations are inserted into the secondary population. The algorithm used is NSGA-II i.e. elitist non dominated sorting genetic algorithm.

Main objective of first phase is to map tasks to available IPs taking task graph as input such that total resource cost and power consumption are minimized. Length of chromosome is a function of number of nodes in the task graph. IPs are chosen from the list of permissible IPs for that task. Single point crossover is used to produce a new generation of solution. For mutation randomly chosen gene is substituted by best permissible IP value for the respective index. The objective of mutation is to assign more tasks to a specific IP to reduce communication. Taking core communication graph as input from first phase and structure of NoC back bone the second phase carries out core-tile mapping with the objective of minimizing communication distance and maximizing throughput. In NSGA II genetic algorithm the each chromosome is defined as a candidate solution whereas each gene represents the application mapped to the tile. Fore cross over a hot core i.e. the IP of which requires maximum communication is identified from the dominant mapping between two chromosomes and is remapped to a randomly selected tile to generate a new chromosome. Mutation is aimed at placing frequently communicating cores together.

The experimental results of [17] show comparison with PBB (Branch and Bound) [14] and PMAP (2-phase mapping algorithm) algorithm. Mutation probability in [17] is kept lower as compared to crossover probability. The benchmark application used in [17] is a modified Motion-JPEG encoder. To emphasize multi-objective optimization in [17] the results are also compared from algorithm with only phase II with those produced when both phases are applied and saving of over 15% energy was observed.

4. EXPERIMENTAL EVALUATION

In [11] comparison results are shown for performance of irregular topology. Figure 1 shows the performance comparison in terms of throughput and flit latency for different routing techniques. Figure 2 to 4 represent the comparative results with respect to mapping methodology proposed in [14] with performance metrics such as average communication energy, average channel communication energy and average communication traffic load. As can be observed from the results of Figure 3 and 4, better traffic distribution is achieved when irregular topology according to the methodology proposed in [11] is used instead of mapping technique of [14].

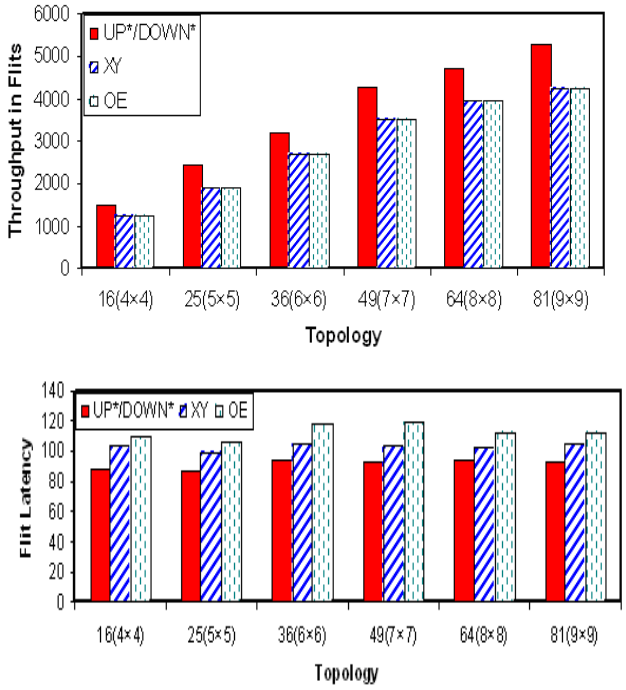

Fig 1: Throughput (in flits) and flit latency (in clocks) with mapping.

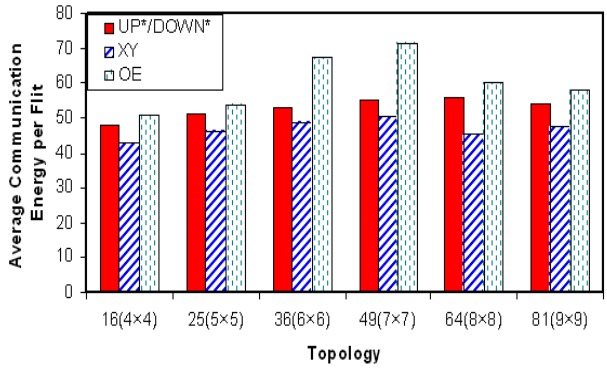

Fig 2: Avg. communication energy consumed by flits with mapping.

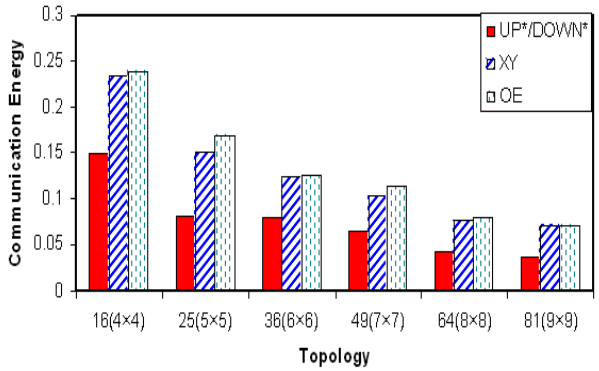

Fig 3: Avg. channel communication energy (pico joules) for mapping.

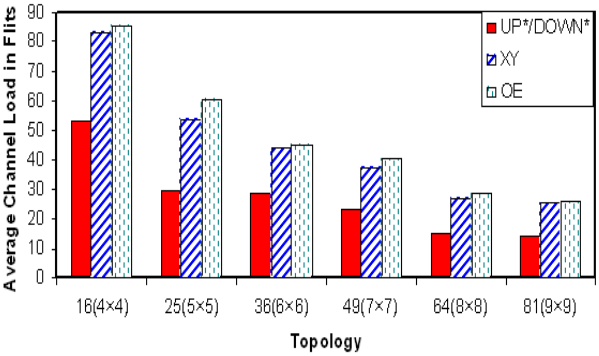

Fig 4: Avg. communication traffic load (in flits) with mapping.

5. CONCLUSION

In this paper we have considered different NoC topology design techniques for efficient NoC synthesis which capture various performance parameters. While most of them try to achieve power and energy optimization through bandwidth and communication energy minimization, there are a few which assume area minimization as of the highest priority and assume an area optimized floorplan as a preprocessing step for customized NoC topology synthesis. In our future work we intend to extend some of the methodologies presented in this paper for enhanced NoC performance.

6. REFERENCES

[1] Dally, W. J., and Towles, B. 2001. Route Packet Not Wires: On-Chip Interconnection Networks. In IEEE Proceedings of the 38th Design Automation Conference (DAC), pp. 684–689.

[2] Bjerregaard, T., and Mahadevan, S. 2006. A Survey of Research and Practices of Network-on-Chip. In ACM Computing Surveys, Vol. 38, March 2006, Article 1.

[3] Duato, J., Yalamanchili, S., and Ni, L. 2003. Interconnection Networks: An Engineering Approach, Elsevier.

[4] International technology roadmap for semiconductors. Semiconductor Industry Association, 2003

[5] Choudhary, N., Gaur, M. S., and Laxmi, V. 2011. Irregular NoC Simulation Framework: IrNIRGAM. In IEEE International conference on Emerging Trends in Network and Computer Communications (ETNCC), Udaipur, India.

[6] Benini, L. 2006 Application Specific NoC Design. DEIS Universit´a di Bologna, IEEE website.

[7] Srinivasan, K., Chatha, K. S. 2005. Isis: a genetic algorithm based technique for custom on-chip interconnection network synthesis. 18th International Conference on VLSI Design held jointly with 4th International Conference on Embedded Systems Design. Kolkata, India, pp: 623-628.

[8] Srinivasan, K., and Chatha, K. S. 2005. SAGA: Synthesis Technique for Guaranteed Throughput NoC Architectures. ASP-DAC 2005.

[9] Srinivasan, K., Chatha, K. S., and Konjevod, G, 2004. Linear Programming Based Techniques for Synthesis of Network-on-Chip Architectures. In Proceedings of ICCD.

[10] Gu, H., Xu, J., and Zhang, W. 2009. A Low-Power Fat Tree-based Optical Network-on-Chip for Multiprocessor System-on-Chip. In Design, Automation & Test in Europe Conference & Exhibition, 2009.

[11] Choudhary, N., Gaur, M.S., Laxmi, V., and Singh, V. 2011. GA Based Congestion Aware Topology Generation for Application Specific NoC. in: Electronic Design, Test and Application (DELTA), 2011 Sixth IEEE International Symposium.

[12] Choudhary, N., Gaur, M.S., Laxmi, V., and Singh, V. 2010. Energy aware design methodologies for application specific NoC. In NORCHIP, 2010.

[13] Jain, L., Al-Hashimi, B.M., Gaur, M.S, Laxmi, V. and Narayanan, A. 2007. NIRGAM: A Simulator for NoC Interconnect Routing and Application Modelling. Proc. DATE 2007.

[14] Hu, J., and Marculescu, R. 2003. Energy-aware mapping for tile-based NoC architectures under performance constraints. ASP-DAC 2003.

[15] Wang, J., Li, Y., Chai, S., and Peng, Q. 2011. Bandwidth-Aware Application Mapping for NoC-Based MPSoCs. In Journal of Computational Information Systems 7:1 (2011) 152-159.

[16] Fekr, A. R., Khademzadeh, A., Janidarmian, M., and Bokharaei, V. S. 2010. Bandwidth/Fault tolerance/ Contention Aware Application-Specific NoC Using PSO as a Mapping Generator. Proceedings of the World Congress on Engineering 2010 Vol I.

[17] Qi, J., Zhao, H., Wang,J., and Li, Z. 2010. A new hierarchical genetic algorithm for low-power network on chip design. In Intelligent Control and Information Processing (ICICIP), 2010 International Conference.

[18] Ascia, G., Catania, V., and Palesi, M. 2006. A Multi-objective Genetic Approach to Mapping Problem on Network-on-Chip. In Journal of Universal Computer Science, vol. 12, no. 4, 370-394.