



APPLICATION MAPPING OF MESH BASED NOC USING EVOLUTIONARY ALGORITHM

JENA R.K.

Institute of Management Technology, Nagpur, India.

*Corresponding Author: Email- rk_jena2@rediffmail.com

Received: January 12, 2012; Accepted: February 15, 2012

Abstract- This paper addresses the problem of mapping of intellectual properties (IPs) cores' on the tiles of a mesh-based NoC using Chaos-multi-objective genetic algorithms (CMGA). Optimization of the energy consumption (Computational and Communicational) and link bandwidth requirement under performance constraints are the objective of the proposed approach.

Keywords- Multi-Objective Optimization, NoC synthesis, Energy consumption, Chaos model.

Citation: Jena R.K. (2012) Application Mapping of Mesh Based NoC using Evolutionary Algorithm. Journal of Information Systems and Communication, ISSN: 0976-8742 & E-ISSN: 0976-8750, Volume 3, Issue 1, pp- 203-206.

Copyright: Copyright©2012 Jena R.K. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Introduction

Network on Chip (NoC) has been proven as an efficient solution for the system-on-chip communication problems. It has been proposed as a solution for the communication challenges like propagation delays, scalability, infeasibility of synchronous communication etc. in a nano scale regime [1-3]. NoC basically addresses the communication-architecture synthesis problem through mapping of cores onto the communication architecture. The problem of application mapping in mesh-based NoC architectures has been addressed in various papers. Hu and Marculescu [2] present a branch and bound algorithm for mapping IPs/cores in a mesh-based NoC architecture that minimizes the total amount of power consumed in communications. De Micheli [4] addresses the problem under the bandwidth constraint with the aim of minimizing communication delay by exploiting the possibility of splitting traffic among various paths. Lei and Kumar [5] presented an approach that uses genetic algorithms to map an application on a mesh-based NoC architecture. However these papers do not solve certain important issues like evaluation model used and the problem relates to the optimization method used.

On the other hand in the above discussed works, most the authors have used exact methods like branch and bound, dynamic

programming algorithms, ILP etc. Although the above stated methods are useful when the design spaces / problem spaces are small. These methods are not efficient for large and complex problem. So in recent years researchers have used heuristic base algorithm to solve the problem. Many NoC mapping problems have been addressed using genetic algorithms. The efficiency of these algorithms is better than the exact methods for complex problem space. Though genetic algorithm based method has shown a lot of its tremendous advantages, it also exposes some limits such as premature convergence and not converging to a global optimization. To resolve this problem, Li and Jiang [14] introduced the concept of Chaos optimization. Chaos movement can no-repeatedly cover all states in a certain range according to its rule. Thus, optimization with chaos variable has without question more advantages than genetic algorithm based random search.

The contribution we intend to make in this paper is to propose a Chaos - multi-objective genetic algorithms (CMGA) approach to solving the NoC mapping problem on a mesh-based NoC architecture. The approach will use to explore the mapping space with the goal to optimize maximum link bandwidth and energy consumption (both computational and communication).

The paper is organized as follows. Section '2' gives the details of chaos optimization algorithm. Section '3' describes the proposed model and Section '4' gives the details problem formulation ideas. Section '5' discusses experimental results. Finally, a conclusion is given in Section '6'.

Mutative Scale Chaos Optimization Algorithm

The simulation results of complex optimization examples show the optimizing efficiency of chaos optimization algorithm over other optimization algorithm such as simulated annealing and genetic algorithm etc. However, it also have been further indicated that the effect of chaos algorithm is very prominent when the searching space is small and not much efficient for large and complex search space. So literature [15] puts forward Mutative Scale Chaos Optimization Algorithm. Mutative Scale Chaos Optimization Algorithm's basic steps are as follows:

1. Algorithm initializing

$$k = 1, k' = 1, x_i(k) = x_i(0), x_i^* = x_i(0), f^* = f(0), a_i(k') = a_i, b_i(k') = b_i .$$

2. Maps the

chaos variable to optimization variable's interval values, and searches according to common chaos optimization algorithm;

$$\text{order } k = k+1, x_i(k) = 4x_i(k)(1 - x_i(k)) \text{ until } f^* \text{ holds the line in a certain steps.}$$

3. Alters the search scale of chaos variable. Where, regulation parameter $c \in (0, 0.5)$ $m x_i^*$ is the current optimal solution.

$$4. \text{ Reverts the optimization variable. } x_i^* = m x_i^* - a_i(k' + 1)(b_i(k' + 1) - a_i(k' + 1)) . \text{ Repeats the search step (2 to 4) with new chaos variable } y_i(k) = (1 - A)x_i^* + Ax_i(k) \text{ (} A < 1 \text{); order } k' = k' + 1 \text{ until } f^* \text{ holds the line in a certain steps.}$$

5. Stop the process after repeating the step (3) and (4) for few times. Here, $m x_i^*$ is the optimal variable and f^* is the optimal solution.

Proposed Model

The proposed NoC communication synthesis problem is shown in the "figure-1". The synthesis task has performed in two phases. The first phase (P-I) is called computational synthesis. The input to P-I of is a task graph. The task graph consists up tasks as vertices and direct edges represent volume of data flowing between two vertices and their data dependencies. The output of P-I is a core communication graph (CCG) characterized by library of interconnection network elements and performance constraints. The second phase (P-II) is basically called as communication synthesis. The input to the P-II communication synthesis problem is the CCG. The output of the P-II is the energy and throughput synthe-

sizes NoC back bone architecture.

In this paper, the problem of mapping the core onto NoC architecture to minimize energy consumption and maximum link bandwidth has been address. Both of the above stated objectives are inversely proportional to each other and NP-hard problems [8]. So, chaos based genetic algorithm is a suitable candidate for solving the multi-objective problem [9]. Experimental result shows that our proposed model is superior in terms of quality of result and execution time in compare to other approaches.

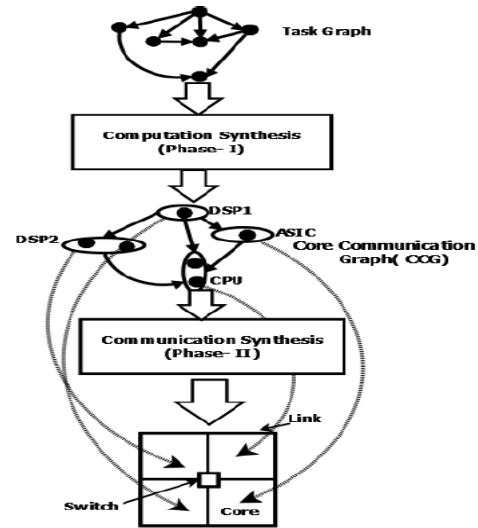


Fig. 1- Mappings for NoC synthesis problems

Energy Model

Energy minimization is the one of the major challenging task in NoC design. In [10], Ye et al. first define the bit energy metric of a router as the energy consumed when a single bit of data goes through the router. In [2], Hu et al. modify the bit energy model so that it is suitable for 2D mesh NoC architecture. They derives mathematical expression for the bit energy consume, when data transfer from switch ' i ' to switch ' j ' is given by

$$E_{i,j,bit} = (h_{i,j} + 1)E_{Sbit} + h_{i,j}E_{Lbit} \text{ (1)}$$

Where E_{Sbit} and E_{Lbit} are the energy consumed in the switches and links respectively. The variable $h_{i,j}$ represent the number of links on the shortest path. As per the expression, the energy consumption is depend on the hop distance ($h_{i,j}$) between switch ' i ' and ' j ' because E_{Sbit} and E_{Lbit} constant.

Note E_{Sbit} is the energy consumption due to switches is depending on the number of ports in the switches. The total communication energy can be derived as ;

$$E_{Comm} = \sum_{i,j} E_{i,j,bit} \text{ (2)}$$

But in our case the total energy (i.e the sum of communication and computation energy) can be computed as:

$$E_{TOTAL} = E_{Comm} + E_{Comp} \text{ (3)}$$

Where, E_{Comm} is the communication energy consumption and E_{Comp} is the computational energy consumption. The following sections discuss the basic ideas of problem formulation using multi-objective optimization paradigm.

Chaos - Multi-objective Genetic Algorithms (CMGA)

In order to deal with the multi-objective nature of NoC problem we have incorporated developed chaos based genetic algorithms at different phases in our model. The algorithm starts with a set of randomly generated solutions by chaos algorithm.

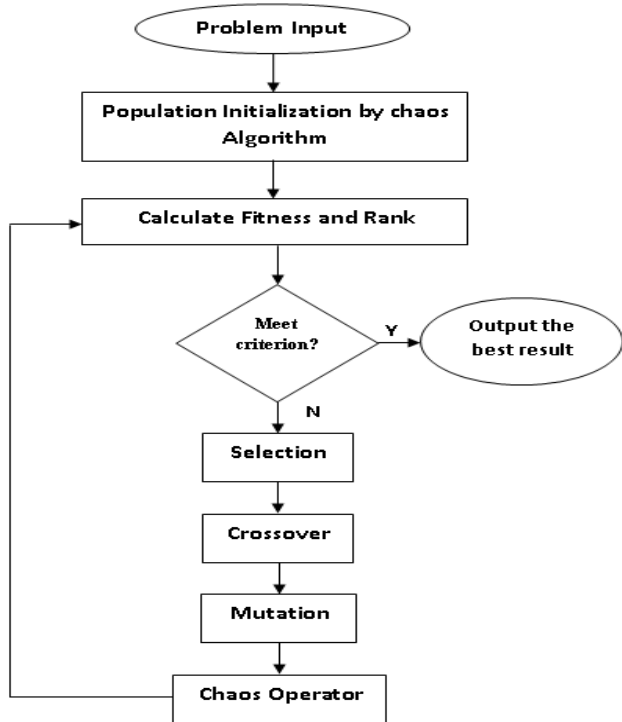


Fig. 2- An Overview of CMGA

The population's size remains constant throughout the GA. Each iteration, solutions are selected, according to their fitness quality (ranking) to form new solutions (offspring). Offspring are generated through a reproduction process (Crossover, Mutation, chaos operator).

In a multi-objective optimization, we are looking for all the solutions of best compromise; best solutions encountered over generations are fled into a secondary population called the " Pareto Archive ". In the selection process, solutions can be selected from this " Pareto Archive " (elitism). A part of the offspring solutions replace their parents according to the replacement strategy. In our study, we used elitist non-dominated sorting genetic algorithm NSGA-II by Deb [11].

Problem Formulation

The algorithm of NoC communication architecture is a hard problem. Our attempt is to develop an algorithm that can give near optimal solution within reasonable time. Chaos based Genetic algorithms have shown the potential to achieve the dual goal. As

shown in "figure-3" and discussed in Section-1, the problem is solved in two phases. The first phase (P-I) is basically a task assignment problem (CMGA-TA). The input to the problem is the TG, we assume that all the edge delays are a constant and equal to Average Edge Delay (AED) [5]. The output of the first phase is a Core Communication Graph (CCG). The task of the second phase is Core-Tile Mapping using genetic algorithm (CMGA-CT). The next sub-section discussed each phases in detail.

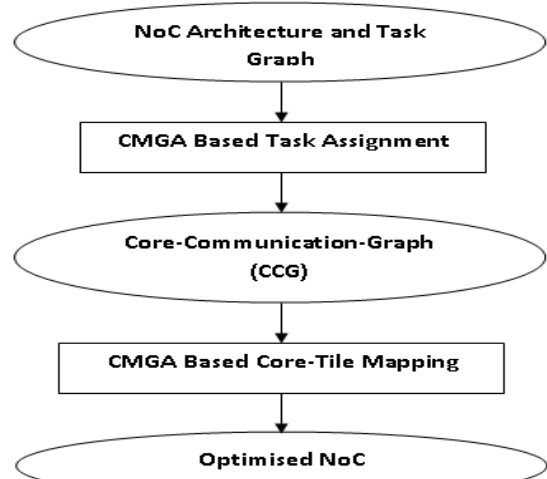


Fig. 3- An Overall design flow

Experimental Results

This section presents the results of our proposed chaotic-multi-objective genetic formulation (CMGA). The final results i.e the results obtained after completion of CMGA-CT are compared with PBB algorithm [2] and PMAP algorithm [4]. "Table 1" shows the bit-energy value of a link and a switch (4x4) assuming 0.18 μm technology.

Table 1- Bit energy values for switch and link

E_{Lbit}	E_{Sbit}
5.445pJ	0.43pJ

In our experiment we consider wormhole routing, which doesn't required buffer in the destination, so we do not include the buffer energy in our model. But, we have small buffers in each RNI to prevent the loss of packets. Again, we consider three random benchmarks (task graph). Benchmark 1,2, 3 contain 16, 18, 14 nodes. After task assignment using TA-GA, the core communication graph of each benchmark consists of less than '9' cores. So, which can be mapped on to a 3x3 mesh NoC architecture. The required bandwidth of an edge connect two different edges is uniformly distributed over the range [0, 150Mbytes].The traffic volume of an edge is uniformly distributed over the range [0, 1G-bits]. "figure-4" shows the maximum link bandwidth utilization of three benchmarks.

It is clear from the figure that our approach (CMGA) saves more than 15% link bandwidth as compare to PMAP and PBB. "figure-5" shows that our approach saves 15%-20 % (on average) of energy consumption.

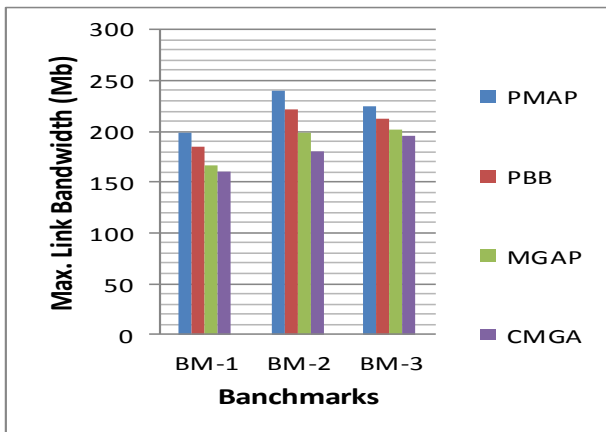


Fig.4- Maximum Link Bandwidth comparisons for three random benchmarks

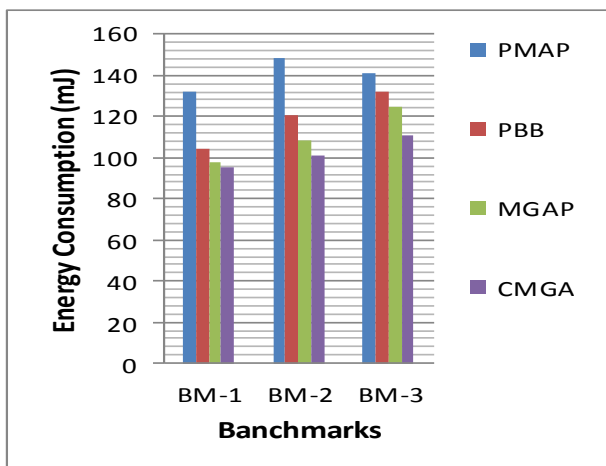


Fig. 5- Energy comparisons for three random benchmarks

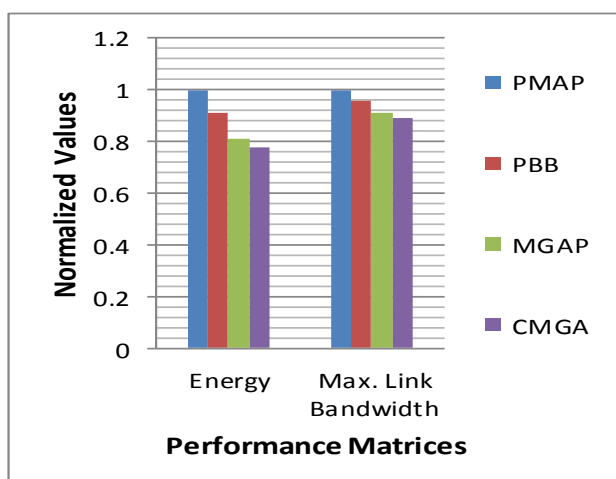


Fig. 6- Maximum Link Bandwidth and Energy comparisons for M-JPEG Encoder

From the above figure (“figure 6”), it is clear that our approach outperform other approaches.

Conclusion

In this paper we have proposed a model for topological mapping of IPs/cores in a mesh- based NoC architectures. The approach uses heuristics based on multi-objective genetic algorithms (NSGA-II) to explore the mapping space and find the pareto mappings that optimize Maximum link bandwidth and performance and power consumption. The experiments carried out with three randomly generated benchmarks and a real application (a M-JPEG* encoder system) confirms the efficiency, accuracy and scalability of the proposed approach. Future developments will mainly address the definition of more efficient genetic operators to improve the precision and convergence speed of the algorithm and many-to-many mapping between core and switches instead of one-to-one mapping. Evaluation will also be made of with the possibility of optimizing the mappings by considering other architectural parameters such as routing topology selection, switch buffer sizes, etc.

Reference

- [1] Zitzler E. and Thiele L. (1999) *IEEE Transactions on Evolutionary Computation*, 4(3), 257-271.
- [2] Hu J. and Marculescu R. (2003) *In Asia & South Pacific Design Automation Conference*.
- [3] Hu J. and Marculescu R. (2003) *DATE*, 688-693.
- [4] Murali S. and Micheli G.D. (2004) *IEEE Computer Society*, 896-901.
- [5] Lei T. and Kumar S. (2003) *Euromicro Symposium on Digital Systems Design*.
- [6] Kumar S. (2002) *ISVLSI*, 105-112.
- [7] Banerjee N., Vellanki P. and Chatha K.S. (2004) *Design, Automation and Test in Europe, Paris, France*, 1250-1255.
- [8] Garey M.R. and Johnson D.S. (1979) *Computers and Intractability: A guide to the theory of NP- completeness*.
- [9] Luca Benini and Giovanni De Micheli (2002) *IEEE Computer*, 70-78.
- [10]Ye T.T., Benini L. and Micheli G.D. (2002) *DAC*, 524-529.
- [11]Deb K. (2001) *Multi-Objective Optimization using Evolutionary Algorithms (John Wiley and Sons Ltd)*.
- [12]Srinivasan K. and Karam S. Chatha, (2005) *18th International Conference on VLSI Design*.
- [13]Pimentel A.D., Polstra S., Terpstra F., Van Halderen A.W., Coffland J.E. and Hertzberger L.O. (2002) *Embedded Processor Design Challenges: Systems, Architectures, Modeling, and Simulation*, 2268 of LNCS, 7-73.
- [14]Li B. and Jiang W.S. (1997) *Control Theory and Applications*, 14, 613-615.
- [15]Zhang T., Wang H.W. and Wang Z.C. (1999) *Control and Decision*, 14, 285-288.