2013 21st Euromicro International Conference on Parallel, Distributed, and Network-Based Processing

ILP-Based Communication Reduction for Heterogeneous 3D Network-on-Chips

Ismail Akturk
Computer Engineering Department
Bilkent University
Bilkent, Ankara, Turkey
iakturk@cs.bilkent.edu.tr

Ozcan Ozturk
Computer Engineering Department
Bilkent University
Bilkent, Ankara, Turkey
ozturk@cs.bilkent.edu.tr

***Abstract*—Network-on-Chip (NoC) architectures and three-dimensional integrated circuits (3D ICs) have been introduced as attractive options for overcoming the barriers in interconnect scaling while increasing the number of cores. Combining these two approaches is expected to yield better performance and higher scalability. This paper explores the possibility of combining these two techniques in a heterogeneity aware fashion. We explore how heterogeneous processors can be mapped onto the given 3D chip area to minimize the data access costs. Our initial results indicate that the proposed approach generates promising results within tolerable solution times.**

***Keywords*-3D, NoC, Heterogeneous, Chip Multiprocessor.**

I. Introduction

International Technology Roadmap for Semiconductors (ITRS) projects that the number of cores will continue to increase [1]. As the number of cores increase, interconnect between these cores becomes a major concern. This is even more pronounced when the number cores is beyond 16 since buses are no longer an option due to physical limitations. Network-on-Chip (NoC) [2] architectures have been proposed to overcome the limitations by using switches and dedicated links between the nodes.

Similarly, 3D Integration is another trend where multiple device layers are stacked together (3D IC) [3]. This trend is driven mostly by greater density, that is, 3D ICs is one of the only ways to meet the demand for increased transistor density. In addition to the density, 3D ICs also provide heterogeneous integration, on-chip interconnect length reduction, modular and scalable design.

NoC architectures have been extended to the third dimension by the help of through silicon vias (TSVs) [4], [5], [6]. 3D NoCs have the potential to achieve better performance with higher scalability and lower power consumption [7], [2]. Most of the related work on 3D NoCs consider homogeneous cores. While, 3D NoCs provide the aforementioned benefits, the best utilization cannot be extracted without including heterogeneity. This is due to the fact that every application (and different parts of an application) has different characteristics. Enabling heterogeneity in 3D NoC architectures will make it possible to match all these various requirements, while keeping energy and heat consumption as minimum as possible. Since heat is one of the most critical issues in 3D ICs, providing heterogeneity has the potential to meet the requirements.

A well known heterogeneous (asymmetric) Chip Multiprocessor (CMP) example is IBM's Cell Processor, where 1 PPU (power processing unit) and 8 SPUs (synergistic processing unit) [8] are combined to perform more efficiently. It was shown that a representative heterogeneous processor using two core types achieves as much as a 63 percent performance improvement over an equivalent-area homogeneous processor [9]. This is mainly due to matching execution resources to application needs effectively.

One of the challenging problems in the context of 3D NoC heterogeneous chip multiprocessor systems is the placement of processor cores within the available chip area. Focusing on such a heterogeneous 3D NoC, this paper explores how different types of processors can be placed to minimize data access costs.

The remainder of this paper is structured as follows. Section II gives the related work on heterogeneous 3D NoCs. Section III discusses the overview of our approach. The details of our ILP (integer linear programming) based formulation are given in Section IV, and an experimental evaluation is presented in Section V. The paper is concluded in Section VI.

II. Related Work

We present the related work in two parts. First, we summarize the related work on 3D NoCs. Then, we explore the related studies on heterogeneous chip multiprocessors.

3D technologies and the motivation for moving from 2D to 3D is explained in []. 3D NoC topologies explored in [2], where they compare 3D NoC to 2D NoC considering physical constraints, such as the maximum number of planes that can be vertically stacked and the asymmetry between the horizontal and vertical communication channels of the network. Li et al. [4] study the L2 design and management in 3D NoC architectures. Ozturk et al. [10] explore how processor cores and data blocks can be placed in a 3D architecture. In [7], authors present a mesochronous communication scheme for 3D NoCs and evaluate its feasibility. Specifically, they analyze the circuit design, the timing properties, the requirements to support flow control across mesochronous

1066-6192/12 $26.00 © 2012 IEEE
DOI 10.1109/PDP.2013.83

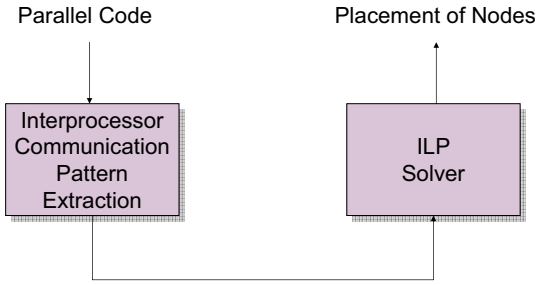

Figure 1. *High level view of our approach.*

links, and the implementation cost of such a scheme after placement and routing. Due to the increasing power density on 3D integrated circuits increasing temperatures becomes a problem. Charles Addo-Quaye [11] presents a genetic algorithm based approach for thermal-aware task mapping and placement for homogeneous 3D NoC designs. Chao et al. [12] presents traffic and thermal aware run time thermal management schemes for three dimensional NoC systems.

Kumar et al. [9] presents potential benefits of heterogeneous chip multiprocessors on different aspects such as overall system throughput and power consumption. Ghiasi et al. [13] presents scheduling techniques on heterogeneous processors on server systems for power management. Blume et al. [14] present a model based exploration method to support design flow of heterogeneous chip multiprocessors. They implement cost models for the design space exploration using several cost parameters such as performance and throughput. Balakrishnan et al. [15] explore the effects of heterogeneity on commercial applications using a hardware prototype. From a hardware perspective, Kumar et al. [16] explore processor design problem for a heterogeneous chip multiprocessor from scratch as processors designed for homogeneous architectures do not sufficiently map to the heterogeneous domain. They study the effects of processor design in terms of area or power efficiency.

III. Overview of Our Approach

High level view of our approach is shown in Figure 1. After a parallelization step, application is passed into a communication analyses module. The analysis module identifies the set of processor nodes that communicate with each other and forwards this information to the ILP solver. ILP solver selects the location of each node in order to minimize the communication cost. Communication cost is estimated based on the 3D distance between the nodes as well as the communication intensity.

Figure 2 illustrates the high level view of a heterogeneous 3D NoC based CMP. While different layers of 3DNoC is connected through TSVs, nodes are connected with network switch/router (represented by R). In the same figure, processor is represented by CPU and memory hierarchy is

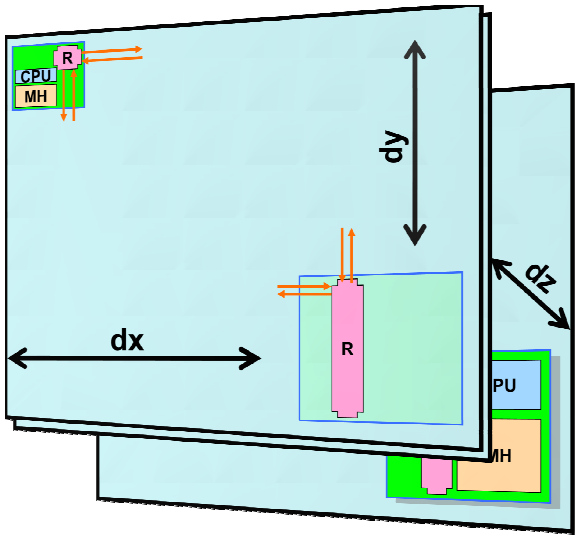

Figure 2. *3D NoC-based CMP architecture.*

represented by MH. Each node is connected to its north, south, west and east via the network switches.

We use L_x, L_y, and L_z to indicate the coordinates of a node in dimensions x, y, and z, respectively. Communication cost is calculated using a Manhattan distance on the respective nodes, that is, $dx = |L_{x1} - L_{x2}|$, $dy = |L_{y1} - L_{y2}|$, and $dz = |L_{z1} - L_{z2}|$. Vertical communication needs to be treated separately from in-layer communication for both latency and bandwidth reasons. Intra-layer communication is expected to be much faster compared to in-layer communication and this needs to be considered in calculating the latencies. Similarly, bandwidth provided by TSVs will be limited and needs to be allocated carefully. We address this issue later in the paper.

IV. ILP Formulation

Our goal in this section is to present an ILP formulation of the problem of minimizing data communication cost of a given application. This is achieved through optimal placement of nodes in a 3D NoC. While overall ILP formulation has more details, for clarity, we will only give important constraints in this section.

Integer linear programming (ILP) is a mathematical model to solve optimization problems using linear objective functions and linear constraints. A special case of ILP is Binary Integer Programming (BIP or 0-1 ILP) where variables are required to be 0 or 1 (rather than arbitrary integers). We use a commercial tool [17] to solve our ILP problem. Table I gives the important constant terms and decision variables used in our ILP formulation. In our ILP formulation, we view the chip area as a 3D grid, and assign nodes into this grid. Therefore, the dimensions of the grid is expressed as D_X, D_Y, and D_Z, respectively. Similarly, for each one of the N nodes, we use SX_c and SY_c to represent a node's

Table I
THE CONSTANT TERMS AND DECISION VARIABLES USED IN OUR ILP FORMULATION. THESE ARE EITHER ARCHITECTURE SPECIFIC OR PROGRAM SPECIFIC. D_Z INDICATES THE NUMBER OF LAYERS IN THE 3D CHIP.

Constant	Definition
N	Number of nodes
D_X	X Dimension of the chip
D_Y	Y Dimension of the chip
D_Z	Z Dimension of the chip
SX_c	X Dimension of node c
SY_c	Y Dimension of node c
$A_{i,j}$	Affinity between nodes i and j
α	Vertical to horizontal communication cost ratio.
Variable	**Definition**
$L^n_{x,y,z}$	Location of node n in x,y, and z dimensions
$Assign^n_{x,y,z}$	Mapping of node n on grid location (x, y, z)
$dx_{i,j,x}$	Distance between nodes i and j in x dimension
$dy_{i,j,y}$	Distance between nodes i and j in y dimension
$dz_{i,j,z}$	Distance between nodes i and j in z dimension

dimensions on a layer. This will be used for mapping and area calculations. Communication load between two nodes is expressed by the affinity matrix $A_{i,j}$, which was explained in the previous section.

We, next, give the decision variables used in our ILP formulation. Location of a node n is captured by L variable. More specifically,

- $L^n_{x,y,z}$: indicates whether node n is on the grid location (x, y, z).

We capture the distance between two nodes by using $dx_{i,j,x}$, $dy_{i,j,y}$, $dz_{i,j,z}$, where they indicate the distances on x-axis, y-axis, and z-axis, respectively. Specifically, we have:

- $dx_{i,j,x}$: indicates whether the distance between nodes i and j is equal to x on the x-axis.
- $dy_{i,j,y}$: indicates whether the distance between nodes i and j is equal to y on the y-axis.
- $dz_{i,j,z}$: indicates whether the distance between nodes i and j is equal to z on the z-axis.

Note that, nodes can potentially use a grid space bigger than one unit, i.e., 1×1. Therefore, we need to use a separate variable to indicate the mapping of the grid space onto different nodes. We use $Assign$ variables to express this.

$$Assign^n_{i,j,k} \geq L^n_{x,y,k}, \forall n, i, j, k, x, y \quad \text{such that } x + SX_n \geq i \text{ and } y + SY_n \geq j. \tag{1}$$

Nodes need to be assigned to a single coordinate on the grid. To satisfy this, we use the following constraint:

$$\sum_{i=1}^{D_X} \sum_{j=1}^{D_Y} \sum_{k=1}^{D_Z} L^n_{i,j,k} = 1, \quad \forall n. \tag{2}$$

Similarly, one coordinate in the grid can be used only for one node. This is enforced by the following constraint.

$$\sum_{i=1}^{N} Assign_{x,y,z,i} = 1, \forall x, y, z. \tag{3}$$

Distances between nodes can easily be captured using the location binary variables. For brevity, we only give the expression for layer-to-layer distance:

$$dz_{i,j,z} \geq L^i_{x_1,y_1,z_1} + L^j_{x_2,y_2,z_2} - 1, \quad z = |z_1 - z_2|. \tag{4}$$

Based on the major constraints given above, we next give our objective function. Our cost function is defined as the sum of the data communication loads in both vertical and horizontal dimensions. More specifically, we denote the total data communication using $Comm_H$ and $Comm_V$ for horizontal, and vertical communication costs, respectively:

$$Comm_H = \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{D_X} A_{i,j} \times dx_{i,j,k} \times k + \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{D_Y} A_{i,j} \times dy_{i,j,k} \times k. \tag{5}$$

$$Comm_V = \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{D_Z} A_{i,j} \times dz_{i,j,k} \times k. \tag{6}$$

Affinity, expressed with $A_{i,j}$, indicates the communication load between the nodes i and j. Therefore, our objective function can be expressed as:

$$\min \quad Comm = Comm_H + \alpha \ Comm_V. \tag{7}$$

Note that, in the objective function given in Expression 7, the difference between horizontal and vertical communication costs is captured by the α parameter which is conservatively set to 0.2 in our baseline implementation. More specifically, accessing a data from a neighboring node on the same layer is five times costlier than accessing a neighbor on a different layer. The α parameter can be exercised and the most suitable value can be used, however we do not discuss this any further.

Note also that, in our ILP formulation, we employ area and distance as two main constraints, whereas performance, energy, and communication bandwidth and other possible constraints are left out. For example, depending on the switch present in a node, bandwidth available to the connected links will be limited. Our ILP formulation, in its current form, does not cover this constraint. However, our formulation can easily be modified to include such constraints. In addition to additional constraints, our ILP formulation can also be modified to optimize for a different objective function instead of data communication cost. However, we do not discuss the details of additional constraints and different objective functions in this paper.

Table II
BENCHMARK CODES USED IN THIS STUDY.

Benchmark	Source	Description	Number of Data Accesses
3step-log	DSPstone	Motion Estimation	90646252
adi	Livermore	Alternate Direction Integration	71021085
ammp	Spec	Computational Chemistry	86967895
equake	Spec	Seismic Wave Propagation Sim.	83758249
mcf	Spec	Combinatorial Optimization	114662229
mesa	Spec	3D Graphics Library	134791940
vortex	Spec	Object-oriented Database	163495955
vpr	Spec	FPGA Circuit Placement	117239027

V. EXPERIMENTAL EVALUATION

To test the effectiveness of our ILP-based approach, we performed experiments using a set of eight array-based applications. Brief descriptions and important characteristics of these applications are listed in Table II. The fourth column of Table II gives the number of data accesses for each application. We tested our approach with four different processors representing different areas and performance characteristics. The ILP solution times varied between 4 minutes and 8 hours, averaging on about 45 minutes. In our base configuration, we used a stack of two device layers connected to one another. We assumed that a single layer is composed of 24 unit areas which can be assigned to NoC nodes. Moreover, we assumed that the vertical communication cost to horizontal communication cost given with α parameter is set to 0.2.

We conducted experiments with four different execution models, namely, *2D-HM*, *2D-HT*, *3D-HM*, and *3D-HT*.

- 2D-HM is the basic execution model where a conventional NoC topology is tested on a single layer with same type of processors. This is the default configuration we compare our results with. Note that, mapping and communication optimizations for this model are implemented using ILP.
- 2D-HT is similar to 2D-HM except that the nodes of NoC can be of different types. Note that, this is an optimal placement scheme for single layer configurations with heterogeneity enabled.
- 3D-HM tries to extend the 2D-HM concept to multiple layers with homogeneous nodes.
- 3D-HT is the integer linear programming based placement strategy for heterogeneous 3D NoCs, wherein different processor cores are placed on several layers optimally. This scheme represents the optimal placement for 3D depending on the communication frequencies of nodes.

Our data communication results are shown in Figure 3. These results are normalized with respect to 2D-HM scheme based on two layers. We see that the overall average reduction in data access costs with 2D-HT and 3D-HM are around 30% and 44%, respectively. On the other hand, 3D-HT scheme reduces the costs by about 54% on average. During our study we simply used the distance between cores to calculate the communication cost without considering the network congestions. We have calculated shortest paths between cores without caring about the congestion. However our ILP solution can be further extended by including congestion and bandwidth related parameters in communication cost function to overcome this issue.

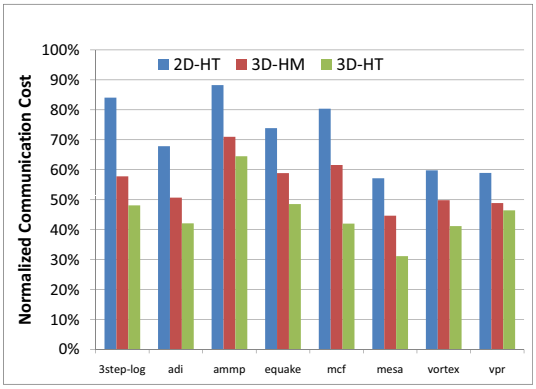

Figure 3. Data communication costs of 2D-HT, 3D-HM, and 3D-HT normalized with respect to 2D-HM.

VI. CONCLUSION

Global interconnect problem has become more important with the increase in the number of processor cores in chip multiprocessing. 3D designs and NoC architectures have been unified as 3D NoCs to overcome the interconnect scaling bottleneck. We try to map heterogeneous processors onto the given 3D chip area with minimal data communication costs. Our initial results indicate that the proposed approach generates promising results within tolerable solution times.

ACKNOWLEDGMENT

This research is supported in part by Turk Telekom under Grant Number 3015-04 and by a Marie Curie International Reintegration Grant within the 7th European Community Framework Programme.

REFERENCES

[1] ITRS, "International technology roadmap for semiconductors."

[2] V. Pavlidis and E. Friedman, "3-d topologies for networks-on-chip," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 15, no. 10, pp. 1081–1090, oct. 2007.

[3] W. Davis, J. Wilson, S. Mick, J. Xu, H. Hua, C. Mineo, A. Sule, M. Steer, and P. Franzon, "Demystifying 3d ics: the pros and cons of going vertical," *Design Test of Computers, IEEE*, vol. 22, no. 6, pp. 498–510, 2005.

[4] F. Li, C. Nicopoulos, T. Richardson, Y. Xie, V. Narayanan, and M. Kandemir, "Design and management of 3d chip multiprocessors using network-in-memory," in *Computer Architecture, 2006. ISCA '06. 33rd International Symposium on*, 2006, pp. 130–141.

[5] S. Murali, L. Benini, and G. De Micheli, "Design of networks on chips for 3d ics," in *Proceedings of the 2010 Asia and South Pacific Design Automation Conference*, 2010, pp. 167–168.

[6] D. Park, S. Eachempati, R. Das, A. K. Mishra, Y. Xie, N. Vijaykrishnan, and C. R. Das, "Mira: A multi-layered on-chip interconnect router architecture," *SIGARCH Comput. Archit. News*, vol. 36, pp. 251–261, June 2008.

[7] I. Loi, F. Angiolini, and L. Benini, "Developing mesochronous synchronizers to enable 3d nocs," in *Design, Automation and Test in Europe, 2008. DATE '08*, march 2008, pp. 1414–1419.

[8] D. Pham, S. Asano, M. Bolliger, M. Day, H. Hofstee, C. Johns, J. Kahle, A. Kameyama, J. Keaty, Y. Masubuchi, M. Riley, D. Shippy, D. Stasiak, M. Suzuoki, M. Wang, J. Warnock, S. Weitzel, D. Wendel, T. Yamazaki, and K. Yazawa, "The design and implementation of a first-generation cell processor," *Solid-State Circuits Conference, 2005. Digest of Technical Papers. ISSCC. 2005 IEEE International*, pp. 184–592 Vol. 1, Feb. 2005.

[9] R. Kumar, D. M. Tullsen, N. P. Jouppi, and P. Ranganathan, "Heterogeneous chip multiprocessors," *Computer*, vol. 38, no. 11, pp. 32–38, 2005.

[10] O. Ozturk, F. Wang, M. Kandemir, and Y. Xie, "Optimal topology exploration for application-specific 3d architectures," in *Design Automation, 2006. Asia and South Pacific Conference on*, 2006.

[11] C. Addo-Quaye, "Thermal-aware mapping and placement for 3-d noc designs," in *SOC Conference, 2005. Proceedings. IEEE International*, 2005, pp. 25 –28.

[12] C.-H. Chao, K.-Y. Jheng, H.-Y. Wang, J.-C. Wu, and A.-Y. Wu, "Traffic- and thermal-aware run-time thermal management scheme for 3d noc systems," in *Proceedings of the 2010 Fourth ACM/IEEE International Symposium on Networks-on-Chip*, 2010, pp. 223–230.

[13] S. Ghiasi, T. Keller, and F. Rawson, "Scheduling for heterogeneous processors in server systems," in *CF '05: Proceedings of the 2nd conference on Computing frontiers*, 2005, pp. 199–210.

[14] H. Blume, H. T. Feldkaemper, and T. G. Noll, "Model-based exploration of the design space for heterogeneous systems on chip," *J. VLSI Signal Process. Syst.*, vol. 40, no. 1, pp. 19–34, 2005.

[15] S. Balakrishnan, R. Rajwar, M. Upton, and K. Lai, "The impact of performance asymmetry in emerging multicore architectures," in *ISCA '05: Proceedings of the 32nd annual international symposium on Computer Architecture*, 2005, pp. 506–517.

[16] R. Kumar, D. M. Tullsen, and N. P. Jouppi, "Core architecture optimization for heterogeneous chip multiprocessors," in *PACT '06: Proceedings of the 15th international conference on Parallel architectures and compilation techniques*, 2006, pp. 23–32.

[17] D. Optimization, "Xpressmp."