Performance Analysis of 3D NoCs Partitioning Methods

Masoumeh Ebrahimi[1,2], Masoud Daneshtalab[2], Pasi Liljeberg[2], Hannu Tenhunen[1,2]
[1]Turku Centre for Computer Science (TUCS), Joukahaisenkatu 3-5, 20520 Turku, Finland
[2]Department of Information Technology, University of Turku, Finland
{ masebr, masdan, pakrli, hatenhu }@utu.fi

***Abstract*— 3D IC design improves performance and decreases power consumption by replacing long horizontal interconnects with short vertical ones. Achieving higher performance along with reducing the network latency can be obtained by utilizing an efficient communication protocol in 3D Networks-on-Chip (NoCs). In this work, several unicast/multicast partitioning methods are explained in order to find an advantageous method with low communication latency. Moreover, two factors of efficiency, unicast latency and multicast latency, are analyzed by analytical models. We also perform simulation to compare the efficiency of proposed methods. The results show that Mixed Partitioning method outperforms other methods in term of latency.**

I. INTRODUCTION

By increasing the number of processing elements in a 2D Multi-Processor Systems-on-Chip (2D MPSoCs), the performance is degraded due to an increase in average interconnect length. Recently, researchers have focused on the communication architecture of 3D NoCs where multiple active silicon layers are vertically stacked, resulting in lower communication latency and power consumption [1]. Furthermore, the choice of unicast and multicast routing protocols can have a large impact on performance and power consumption [2]. The routing protocols in NoC and MPSoC can be unicast or multicast. Unicast communication is defined by sending a message from a source node to a single destination, while in multicast communication a message is sent from a source node to an arbitrary set of destinations. For preventing deadlock in networks which support unicast/multicast routing, the Hamiltonian Path method was introduced. Different partitioning methods can be defined in networks based on the Hamiltonian Path in order to reduce the number of hop counts taken by messages. In this work, we present three partitioning methods along with their analytical models for evaluating unicast/multicast latency. We also perform simulation to compare average unicast/multicast latency of three proposed methods.

II. HAMILTONIAN PATH

The Hamiltonian Path method guaranties that the network will be free of deadlocks for unicast and multicast traffic. The Hamiltonian Path visits each node exactly once along the path. As shown in Fig 1, for each node a label is assigned from 1 to N-1 in which N is the number of nodes in the network. Several Hamiltonian Paths can be considered in a mesh topology. In 3D a×b×c mesh, each node is presented by the ordered triple (x,y,z). The following equations show one possibility of assigning the labels which we utilize in this work:

$$L(x,y,z)=\{(a\times b\times z)+(a\times y)+(x)\} \qquad where\, z: even, y: even$$
$$L(x,y,z)=\{(a\times b\times z)+(a\times y)+(a-x-1)\} \qquad where\, z: even, y: odd$$
$$L(x,y,z)=\{(a\times b\times z)+(a\times(b-y-1))+(a-x-1)\} \qquad where\, z: odd, y: even$$
$$L(x,y,z)=\{(a\times b\times z)+(a\times(b-y-1))+(x)\} \qquad where\, z: odd, y: odd$$

Two directed Hamiltonian Paths (or two subnetworks) can be constructed by the labelling. The high channel subnetwork starts at node 0, and the low channel subnetwork ends at node 0. In case the label of the destination node is greater than the label of the source node, the routing always takes place in the high subnetwork; otherwise it takes place in the low subnetwork. The destinations are placed into two groups. One group contains all the destinations that could be reached using the high channel subnetwork, and the other contains the remaining destinations that could be reached using the low channel subnetwork.

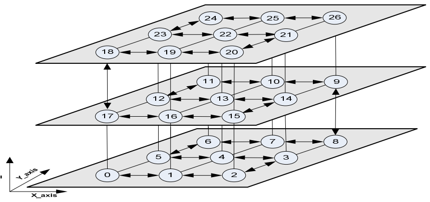
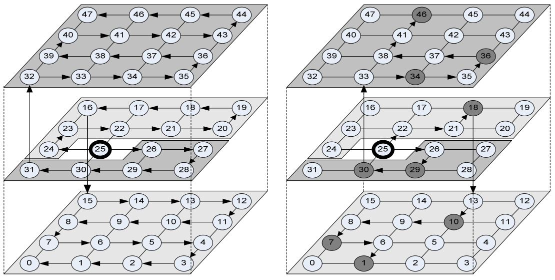


Fig 1. Hamiltonian Path

III. DUAL-PATH PARTITIONING (DPP)

In this scheme, the network is partitioned into high and low channel subnetworks as shown in Fig 2-a. The high channel (low channel) subnetwork contains all directional channels with nodes labeled in ascending order (descending order). In this method all destination nodes are split at most into two disjointed groups: High group and low group. The high group (low group) consists of all destination nodes with the higher labels (lower labels) than the source node. All destination addresses in high and low groups should be placed into two separate messages and routed via high and low channel subnetworks, respectively. Additionally, the destinations should be sorted within each message in the right order in which they are visited in the path. In order to achieve this goal, destinations in the high group should be sorted in ascending order and other destinations in descending order. Fig 2-b (for simplicity the vertical via-s were removed) shows a minimal Dual-Path routing example where the node with the label 25 generates a multicast message toward several destinations with labels {1,7,10,18,29,30,34,36,46}. At first the destinations will be divided into G_H={29,36,34,30,46} and G_L={7,1,10,18}. After the sorting in each group, the sorted destinations in group1={29,30,34,36,46} and group2={18,10,7,1} should be put into message1 and message2 and deliver to the high and low channel subnetworks, respectively.



(a) (b)

Fig 2. (a) Dual-Path partitioning (b) example of minimal Dual-Path routing

A. *Network Latency for unicast and multicast messages*

The analytical models are provided for unicast/multicast messages supposing the zero-load latency [1]. Considering the zero-load latency, a packet never contends for network resources with other packets and therefore, a lower bound on the average latency of a packet traversing through the network can be calculated. Under this assumption, the performance of a routing algorithm can be measured primarily by the number of hop counts required for delivering the message from a source node to any destination [3]. Since unicast and multicast messages can be routed inside the network simultaneously, average number of hops is required to be measured for both unicast and multicast messages. Assuming the dimension-order routing for unicast messages, the average number of hops for 3D a×b×c mesh is [1]:

$$AvgUnicast = \frac{abc(a+b+c) - c(a+b) - ab}{3(abc)}$$

Each multicast message contains several destinations, and the number of hop counts taken by a multicast message depends on the number of destinations and their locations. Therefore, calculating the average number of hops for multicast messages cannot be easily analyzed and modeled analytically. In order to find an analytical formula to calculate the latency of multicast messages, we assume the messages travel within the longest path to reach the last destination without considering intermediate destinations. As a result, the factor is determined as an average of maximum hop counts between each source node and destination. This factor for source node j in a×b×c network is taken by:

$$AvgMaxMulti_{DPP} = \frac{1}{abc}\left(\sum_{i=1}^{i=j} i + \sum_{i=j+1}^{i=n-1}(i-j)\right)$$

And the average maximum hop counts for the whole network is:

$$AvgMaxMulti_{DPP} = \frac{1}{abc}\sum_{j=0}^{j=n-1}\frac{1}{abc}\left(\sum_{i=1}^{i=j} i + \sum_{i=j+1}^{i=n-1}(i-j)\right) = \frac{(abc)^2 - 1}{3abc}$$

IV. DUAL BASED COLUMN-PATH PARTITIONING (DBCPP)

In this method, similar to the Dual-Path partitioning method, the network is partitioned into high and low subnetworks; destination nodes are divided into high and low groups and then sorted in each group. In the next step, each subnetwork (group) will be partitioned further in which the nodes (destination nodes) with the same *x* value will be put in the same subnetwork (group). An example is shown in Fig 3-a. In comparison with DPP method, this scheme reduces the overall path length considerably but the drawback is higher start up latency, possibly due to generating more messages.

A. Network Latency for unicast and multicast messages

The average unicast latency is similar to DPP method due to utilizing dimension-order routing. For multicast messages, the packet can choose both *x* and *y* dimensions in a layer that source node is located, however, for other layers, only *y* dimension can be used. In order to obtain the average maximum multicast hop counts in DBCPP method, the network can be imagined as 2D a×b′ mesh network where b′ dimension value is equal to b×c. Now, dimension-order routing can be utilized for each message and therefore, an average of maximum hop counts for multicast messages in DBCPP method will be:

$$AvgMaxMulti_{DBCPP} = \frac{(abc-1)(a+bc)}{3(abc)}$$

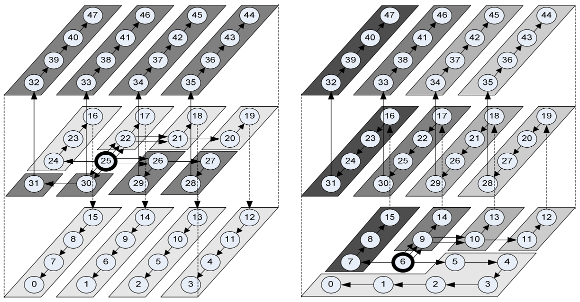
V. MIXED PARTITIONING (MP)

Depending on source node location, the number of destinations generated by a core and level of congestion in the network, different methodologies have different levels of efficiency. The basic idea behind the mixed partitioning method is to utilize either DPP or DBCPP method in the same network. As illustrated in Fig 3-b, the high subnetwork which covers considerably more nodes can be partitioned further using DBCPP method and the low subnetwork employs DPP method. Since the intersection of all partitions is always empty and all paths are followed in ascending or descending order, no deadlock will occur. In comparison with two previously mentioned methods, this scheme presents better distribution of nodes among groups with desirable start up latency.

A. Network Latency for unicast and multicast messages

The average unicast latency for MP method is similar to DPP and DBCPP methods due to utilizing dimension-order routing. The multicast message latency is similar to DBCPP method except high and low layers. In these layers, the calculation is based on both DPP and DBCPP methods. Therefore, average maximum hop counts for MP method is:

$$AvgMaxMulti_{MP} = \frac{(abc-1)(a+bc))}{3(abc)} - \frac{(ab-1)(a+b)}{3(abc)} + \frac{(ab)^2 - 1}{3(abc)}$$



(a) (b)

Fig 3. (a) Dual based Column-Path Partitioning (b) Mixed partitioning

VI. RESULTS

The analytical results in Table 1 indicate that an increase in the network size results in an increase in the average unicast latency. Also, as obtained by the analytical models, the average maximum multicast latency of DBCPP method is less than MP method. That is because of utilizing DPP in MP method that causes the network to be divided into fewer partitions containing more nodes. However, according to simulation results, MP method performs better in term of average multicast latency.

Table 1. Analytical results of DPP, DBCPP and MP

Method	NoC Size	AvgUnicast Latency	AvgMax MulticastLatency	NumOf Partitions
DPP	3×3×3	2,67	8,99	2
DPP	4×4×3	3,39	15,99	2
DBCPP	3×3×3	2,67	5,13	8
DBCPP	4×4×3	3,39	5,22	8
MP	3×3×3	2,67	5,53	5 or 8
MP	4×4×3	3,39	6,16	5 or 8

In addition to analytical model, we developed a cycle accurate wormhole 3D-mesh NoC simulator to compare and assess the efficiency of proposed methods. In a simulation, we have fixed the mesh size to 4*4*3, the message length to 5 flits, the flit width to 32 bits, the number of destinations to 16 nodes for multicast messages, and the traffic pattern is random. Additionally, 70 % of injected messages are multicast and 30% are unicast. For simulation results, the routing algorithm is considered to be deterministic; however, adaptive routing can be easily extended to all of the proposed method from 2D NoC by utilizing HAMUM method [4]. According to the results, near the saturation rate, MP has 26% and 41% lower average latency than DBCPP and DPP, respectively.

SELECTED REFERENCES

[1] V. F. Pavlidis and E. G. Friedman. 3-D topologies for networks-onchip. IEEE Transactions on Very Large Scale Integration Systems, 15(10):1081–1090, 2007.

[2] M. Daneshtalab, M. Ebrahimi, S. Mohammadi, A. Afzali-Kusha, “Low distance path-based multicast algorithm in NOCs,” IET (IEE) Special issue on NoC, Vol. 3, Issue 5, pp. 430-442, 2009.

[3] W.J. Dally and B. Towles, Principles and Practices of Interconnection Networks. Morgan Kaufmann, 2004.

[4] M. Ebrahimi, M. Daneshtalab, P. Liljeberg, H. Tenhunen, "HAMUM – A Novel Routing Protocol for Unicast and Multicast Traffic in MPSoCs", Proc. of PDP conference, IEEE Press, pp. 525-532, 2010.