A Traffic Equilibrium Mapping Method with Energy Minimization for 3D NoC-Bus Mesh Architecture

Taotao Zhang, Ning Wu, Fang Zhou, Lei Zhou, and Xiaoqiang Zhang

***Abstract*—Flit transmission in the vertical direction of 3D NoC (Network-on-Chip) -Bus mesh architecture just needs one hop and consumes less energy. To take advantage of that and to solve the defect of poor heat dissipation, a new traffic equilibrium and energy minimization mapping method for the architecture is proposed. The proposed method adjusts traffic among buses or layers, which provides a natural condition for heat balance in the mapping step, and ensures lower energy consumption which benefits from the characteristics of the architecture. Experimental results show that energy consumption is decreased and tradeoffs can be made between energy minimization and traffic equilibrium.**

***Index Terms*—3D NoC-Bus, map, traffic equilibrium, energy minimization**

I. Introduction

THE mapping process of 3D NoC (network-on-chip) has a great influence on the traffic distribution and power consumption in the chip. Traffic balanced distribution is conducive to enhance the system's communication performance and stability. So, to design a balanced traffic distribution and little power consumption mapping algorithm is of great help to improve network performance of 3D NoC.

In 3D integrated circuits (ICs), by using through silicon vias (TSVs), multiple planar device layers are stacked and bond together, as shown in Fig. 1. As the gap between each layer is only tens of micrometers, the length of TSV is much smaller than that of the horizontal link, which cases little resistance and capacitance in TSVs. So when transmitting the same amount of data, TSVs consume much less energy than horizontal links [1].

To make better use of the advantages of TSV in 3D ICs,

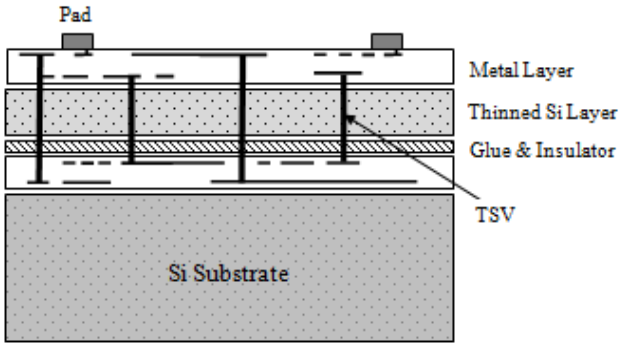


Fig. 1. 3D integration using TSVs

the vertical interconnect utilizes a wide, low-latency multi-drop bus, which forming 3D NoC-Bus hybrid architecture [2]. The cores in the same stack are controlled by a centralized Arbiter, in which flit can be accessed in only one hop.

However, 3D ICs greatly increased device density, which introduces serious thermal issues. As high temperature can cause the chip unstable and reduce the chip's lifetime, thermal issue becomes a major challenge for 3D integration. In the literature, many methods have been proposed, such as incorporating thermal vias into integrated circuits to mitigate thermal issues by lowering the thermal resistance of the chip [3], planning thermal via locations during floorplanning stages [4], cooling each level in the 3D stack by microfluidic interconnects [5], scheduling dynamic thermally-aware job for 3D systems [6], designing a topology uniformly distributed [7], designing a thermal aware mapping algorithm [8], and targeting minimum signal TSVs [9], etc.

Works described above didn't involve the characteristics of 3D NoC-Bus mesh architecture, just like flit can be accessed in only one hop through bus which makes the mapping more flexible, and TSVs consume much less energy than horizontal links. Some of our initial work has been presented in [10] to optimize the mapping result of 3D Bus-NoC Mesh architecture. However, it didn't consider the buses' position of the whole structure and the Manhattan distance among the communication nodes. In this paper, through distributing traffic uniformly to make an even heat distribution, or restricting tasks within the same stack and replacing locations of buses to reduce transmission energy, a new mapping method for 3D NoC-Bus mesh architecture is proposed.

Manuscript received January 20, 2015. This work was supported by the National Natural Science Foundation of China under Grant (No.61376025, No.61106018), the Industry-academic Joint Technological Innovations Fund Project of Jiangsu under Grant (No.BY2013003-11), the Funding of Jiangsu Innovation Program for Graduate Education under Grant (No.KYLX_0273), and the Fundamental Research Funds for the Central Universities.

Taotao Zhang, the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing 210016, China (email: taotzhang@163.com).

Ning Wu, the College of Electronic and Information Engineering, NUAA, Nanjing 210016, China (email: wunee@nuaa.edu.cn).

Fang Zhou, the College of Electronic and Information Engineering, NUAA, Nanjing 210016, China (email: zfnuaa@nuaa.edu.cn).

Lei Zhou, the College of Information Engineering, Yangzhou University, Yangzhou 225009, China (email: tomcat800607@126.com).

Xiaoqiang Zhang, the College of Electronic and Information Engineering, NUAA, Nanjing 210016, China (email: zxq198111@qq.com).

II. Principles of the Proposed Method

To slove the problem, two definitions, an application core graph *ACG(V,E,W)* and a 3D NoC-Bus Mesh architecture graph *NBAG(N,B)* , are set up as inputs as following:

ACG(V,E,W) is a weighted directed graph, in which V is a set of IP cores, E is a set of communications and W is a set of communication volume belong to each directed edge. Each vertex $v_i \in V$ represents an IP core, each directed edge $e_{i,j} \in E$ represents the communication from IP core v_i and v_j, and each $w_{i,j}$ denotes the communication volume of the edge $e_{i,j}$.

NBAG(N,B) is an undirected graph, where N is a set of nodes, and B is a set of Buses. Each node $n_i \in N$ is specified with the coordinate (x_i, y_i, z_i) to describe the position, and the position of each bus $b_i \in B$ is specified with the coordinate (x_i, y_i). To make the mapping method more universality, we choose the popular 3D NoC-Bus Mesh as the target architecture.

A. Energy model

We assume $P_{horizontal}$ as the consumed energy transmitting a bit of data per unit length in the horizontal layer, $P_{vertical}$ as the energy in the vertical direction, and P_{route} as in the routing unit. So the energy consumed in a route unit by transmitting a bit data can be given by

$$P = (P_{horizontal} \times d_h + P_{vertical} \times d_v + P_{route}) \times w \quad (1)$$

where w denotes the data flow passing the route unit per unit time, d_h and d_v denotes the links' length in the horizontal direction and vertical direction respectively. The detailed calculation of $P_{horizontal}$ and $P_{vertical}$ can refer to [1].

B. Optimization model

As the heat and traffic are positively correlated and the inputs include communication volume, so we can restrict the heat by controlling the traffic. Through adjusting each node's location in bus, a map of traffic uniform distribution in horizontal layer can be found which will also generate a uniform heat distribution in horizontal layer. So the key constraint of heat is restricting the traffic on the buses.

So the optimal mapping problem can be described as

$$\begin{cases} Min\{W_{horizontal}[Min(S_L((6)\ or\ (7)))]\} & (2) \\ S_L = \sum_{i=1}^{L}(\sum_{j=(i-1)\times(N_L+1)}^{i\times N_L} w_j - E_L)^2 & (3) \\ E_L = \sum_{i=1}^{L\times N_L} w_i / L & (4) \\ W_{horizontal} = \sum_{i=0,j=0}^{n} w_{i,j} \times (|x_i - x_j| + |y_i - y_j|) & (5) \\ Max(\sum_{i=1}^{m} W_{bus\_i}) \ \&\ Max(W_{bus\_i}) \le \alpha \bullet W_{b\_\max} & (6) \end{cases}$$

$$Min((\sum_{i=1}^{m} W_{bus\_i} - \sum_{i=1}^{m} W_{bus\_i}/m)^2) \quad \& \ Max(W_{bus\_i}) \le \alpha \bullet W_{b\_\max} \quad (7)$$

where $W_{horizontal}$ and W_{bus_i} are the total communication volume in the horizontal layer and bus respectively, W_{b_max} is the max total communication can be found in the bus combinations, m is total number of buses, α is a percentage constant, S_L is the variance among the communication volume of the layers, and E_L represents the average value of the layers, N_L is the node number each layer. Condition (2) means the proposed method minimizes the energy consumption by minimize the total communication volume in the horizontal layer with a uniform distribution among layers which is under the condition (6), (7) of communication in buses, of which the communication concentrates on buses or evenly distributes on buses under the limit that each bus has a maximum traffic.

C. Mapping Method

As is stated above, this mapping method's mission is mapping each IP core of *ACG* onto a network node of *NBAG* and ensures minimum energy consumption under the constraint of traffic equilibrium.

Unlike the traditional 3D NoC architecture, 3D NoC-Bus mesh architecture uses bus communication to replace the network communication in the vertical direction. Since the architecture only needs one hop communicating on the bus, it greatly improves latency performance of system in the 3D NoC-Bus mesh architecture. And the character of single hop communicating on the bus provides more choices for mapping the nodes of the task graph to the 3D NoC-Bus mesh architecture. Because changing the position of the nodes on the same bus doesn't affect the delay of the communication between them and has a small impact on the overall structure.

As is calculated, bit transmission cost the horizontal link 0.127 pJ and the TSV 9.56×10^{-3} pJ [1]. Bit transmission energy for TSV is only 7.5% of that of horizontal link. So it can reduce plenty of communication energy if aggregates tasks within the same stack. But the more tasks cluster in the same stack, the more energy density of this stack increase, which will be likely to generate a hot spot. We first map nodes to the buses because it has the greatest impact on energy consumption. So this method takes three steps to map the task graph to the 3D NoC-Bus mesh architecture.

First, map the nodes of *ACG* to the buses of *NBAG* which need to break the tree structure of *ACG* into links using backtracking algorithm. It starts with the root node of the tree and goes to the child node of current node. If current node that is a cross point has multiple nodes, go small node first. Until reach the end of the link, go back to the cross point to find another link. Traversing all the links in the steps of the number of layers of 3D NoC-Bus mesh architecture, which is also the number of nodes on the bus, and the bus units can be found. Combining those units to make combinations of buses without a same node and screening out the combinations that meet the requirements.

Second, map the nodes on the buses to different layers of *NBAG*. Because of a permutation and combination of nodes on the buses is a mapping result, there will be many results if there are many nodes on a bus. We use genetic algorithm to generate new results and filter the bad results generation to generation. Select the result meet the needs best in the last generation as the last mapping result.

Third, replace the buses' locations to find the one in which the transmission energy is the least. According to the two above steps, the bus and layer the nodes belong to have been identified. Transforming the locations of buses will change the Manhattan distance of the nodes, therefore, it will affect the transmission energy of horizontal layers.

III. Task Mapping

A. Task Graph Analysis

In order to fully describe the mapping algorithm in this paper, we use Task Graphs for Free (TGFF) [9] to generate a 20-node task graph randomly, as shown in Fig. 2.

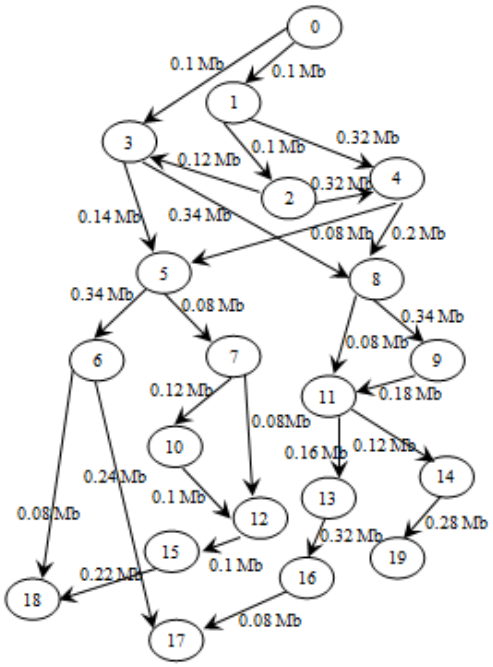


Fig. 2. 20-node task graph

Task graph is an acyclic directed graph derived from a special foreseeable application. In the graph, every node represents a task to be assigned to the core. Nodes connect to each other if they have communication requirement with other nodes. The weight on the edge denotes the communication data volume between the corresponding nodes.

According to the number N of nodes in the task graph and the layers L need to be generated, an n×n×L 3D NoC-Bus hybrid architecture can be finalized via the following inequation and equation:

$$\begin{cases} (n-1)^2 < N_L \le n^2 & (8) \\ N_L \ge N / L & (9) \end{cases}$$

N_L is the minimum integer bigger than N/L, which denotes the maximum nodes in the task graph can be allocated in a layer. If $N_L < n^2$, other n×n×L - N nodes should be added to the n×n×L 3D NoC-Bus hybrid architecture.

When the task graph in Fig. 3 is mapped to the 3×3×3 3D NoC-Bus hybrid architecture, 2 layers of the architecture will be allocated 7 nodes, and 6 nodes in the remaining layer. In addition, it takes 7 additional nodes to fill the 3×3×3 architecture. Since the communication between the nodes are very different, how to layout this nodes will make great impact on system performance. Considering transmitting in TSVs not only can reduce much energy than in horizontal links [1] but also has a greater influence on the heat dissipation [11]-[13], the promised algorithm maps the nodes in the stack first and then maps the nodes in the horizontal layers.

B. Mapped to the Stacks

To assign the nodes in the task graph to the stacks, backtracking algorithm is utilized. The proposed method break up the tree structure into several links according to the parent-child relationship nodes.

First, look for links from the root node and go the path of small node preferentially when come across a bifurcate point. Until reach the end node, then return to the bifurcate point and find other links recursively.

Doing in this way can ensure that the correlative nodes are allocated in the same stack. As shown in Fig. 3, to map the task graph in a 3×3×3 3D NoC-Bus architecture, 20 nodes should be mapped to 7 stacks, one of which contains two nodes and each of the other contains 3 nodes. Done in accordance with the above, the task graph can be broken into 32 links, as shown in Table I.

Second, after finding out all the links, traverse the nodes on the links using L as a unit. Put the units that don't contain the same node together as a combination of stacks. If the number of units that don't contain the same node is smaller than the number of stacks we need, find other units using L - 1 as a unit in the rest nodes, which are not in the units, as shown in Fig. 3.

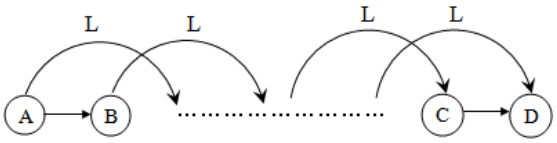


Fig. 3. Method of finding the L units in a link

A, B, C and D represent the nodes in a link of the task graph. L is the number of layers (the number of nodes a stack includes).

As is calculated, bit transmission cost the horizontal link 0.127 pJ and the TSV 9.56×10^{-3} pJ [1]. Bit transmission energy for TSV is only 7.5% of that of horizontal link. So it can reduce plenty of communication energy if aggregates tasks within the same stack. But the more tasks cluster in the same stack, the more power density of this stack increase, which will be likely to generate a hot spot.

This paper maps the nodes of the task graph in two extreme ways, the optimal energy consumption and the optimal balanced distribution of traffic. The former only considers aggregate as many tasks as we can in the same stack to save maximum communication energy and the latter just takes an

average allocation in the stacks to make a uniform distribution of heat. According to the needs, considering them both is a good method to make a tradeoff between energy consumption and heat dissipation. They can be gotten at the same time due to they are not an inverse relationship.

TABLE I
LINKS FOUND FROM THE TASK GRAPH

Link	No.
0-1-2-3-5-6-17	1
0-1-2-3-5-6-18	2
0-1-2-3-5-7-10-12-15-18	3
0-1-2-3-5-7-12-15-18	4
0-1-2-3-8-9-11-13-16-17	5
…	...
0-3-8-9-11-14-19	30
0-3-8-11-13-16-17	31
0-3-8-11-14-19	32

1) Optimal Energy Consumption

6 stacks of 3 nodes can be found from the links in Table I, and the rest two nodes is allocated in the last stack. As the proposed method computed, there are 241 kinds of combinations. When only considering energy consumption, aggregating tasks in the same stack, using F_{stacks} as the degree of aggregating tasks, result can be reached when F_{stacks} is the maximum of the combinations as shown in Fig. 5. The "*" in Fig. 4 represents the node additional added.

$$F_{stacks} = \sum_{i=1}^{n} F_{vi} \tag{10}$$

F_{vi} represents the communication data volume of each stack.

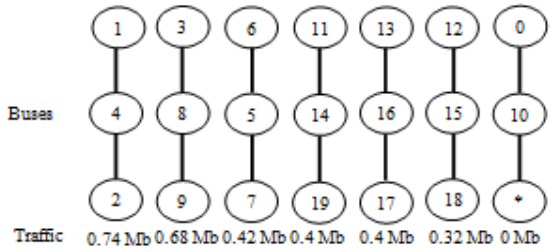


Fig. 4. Map of stacks of optimal energy consumption

2) Optimal Balanced Distribution of Traffic

To allocate the tasks averagely to the 7 stacks, which can obviously form a uniform distribution of heat to solve the thermal issues, we use S_v, the variance among the communication data volume of the stacks, as a yardstick to measure the degree of uniformity of traffic in each stack.

$$S_v = \sum_{i=1}^{n} (F_{vi} - E)^2 \tag{11}$$

E denotes the average value of all the n stacks. Via this equation, the result of optimal balanced distribution of traffic can be found when S_v is the minimum of the combinations, as shown in Fig. 5.

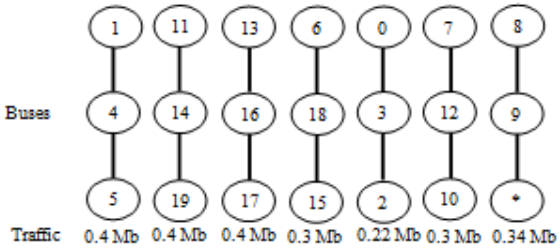


Fig. 5. Map of stacks of optimal balanced distribution of traffic

C. Mapped to the Horizontal Layers

Except the two optimal combinations of buses above, we can also find other combinations that meet our requirements and unite the result mapping to the horizontal layers to find the best mapping result among them.

Due to the transmission between nodes on the bus just need a single hop, it has little impact how to arrange the nodes on the stack. So the nodes on the bus can be sorted casually and what the proposed method need to do is finding out which arrangement of the nodes on the buses is the best. Genetic algorithm is used to find the best permutation we need. A model of chromosome is created in this algorithm, as shown in Fig. 6.

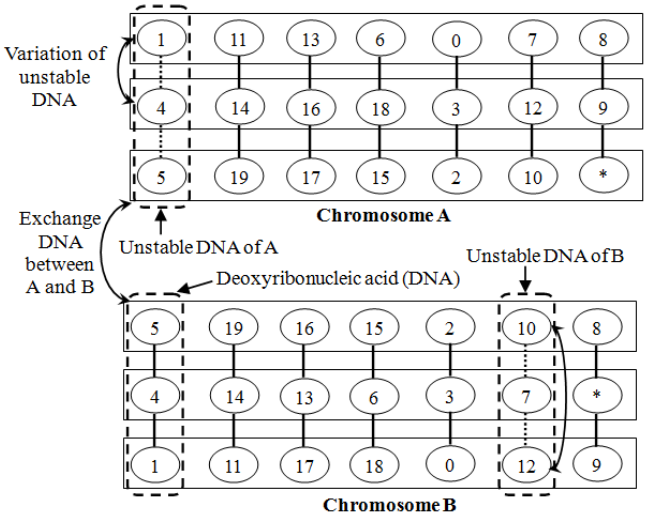


Fig. 6. The way of chromosomal heredity of the genetic algorithm of the proposed method

The length of the chromosomal which has multilayer structure is equal to the number of bus need to be mapped. A bus represents a deoxyribonucleic acid (DNA) in the chromosome. Every chromosome has only one unstable DNA (connected with dashed lines in Fig. 7) that is susceptible to mutation. A mutation or exchange of chromosomes is equivalent to change the nodes sort on the buses.

To choose the best chromosomes we need, a list of total communication data volume belong to each node is created. Encode each DNA in the chromosomes to sum the total communication data volume of each layer. To achieve a balanced traffic of each layer, we use S_L, variance among the

communication volume of the layers, to measure the degree of uniformity of traffic in each layer.

$$\begin{cases} S_L = \sum_{i=1}^{L} \left(\sum_{j=(i-1)\times(n+1)}^{i\times n} F_j - E_L \right)^2 & (12) \\ E_L = \sum_{i=1}^{L\times n} F_i / L \times n & (13) \end{cases}$$

L represents the number of layers of chromosomes. E_L represents the average value of the layers. F_j denotes the total communication volume flowing through the node j. j is recoded according to the structure of chromosomes. For example, F_{10} of Chromosome A in Fig. 6 represents the total communication volume of node 16 of the task graph in Fig. 3, which is 2Mb.

Unlike the very low probability of gene mutation in the nature, here the mutation should have high probability to keep the diversity each generation. Then sort chromosomes in ascending order by S_L and keep the previous 50 chromosomes that meet the needs and reset all the unstable DNAs in the chose chromosomes to ensure diversity each generation. Until the last generation, choose the first chromosome as the best mapping result of horizontal layers.

In the experiment, the initial population size is 100, the crossover rate is 0.8, the mutation rate that is big here is 0.5, the maximum evolution generation is 500 and the length of chromosome is 7. The mapping results of horizontal layers of optimal energy consumption and optimal balanced distribution of traffic, which respectively are {{3, 1, 6, 11, 13, 12, 0};{8, 4, 5, 14, 16, 15, *};{9, 2, 7, 19, 17, 18, 10}} and {{1, 11, 13, 6, 0, 7, 9};{4, 14, 16, 18, 3, 12, *};{5, 19, 17, 15, 2, 10, 8}}, in which the nodes in the same set belong to the same layer, can be reached with the parameters being set above.

D. Replace the bus location

After the two steps above, the bus and the layer a node belongs to is identified. The 3D NoC-Bus hybrid architecture has basically taken shape. But we can still optimize the mapping result by changing the position of the buses. When the position of the buses transformed, the Manhattan distance between the nodes which has impact on the transmission power of the horizontal layers changed too. So the mapping result of the minimum power consumption of horizontal transmission can be found by transforming the position of the buses.

The mapping result of the optimal balanced distribution of traffic can be transformed as shown in Fig. 7. Every node has a bus index to make sure which bus it belongs to. Use the coordinate to compute the Manhattan distance between two nodes. For example, the Manhattan distance between node 0 and node 5 is $|0 - 1| + |0 - 2| = 3$.

Here, we reduce the transmission power by reducing the total communication data volume in the horizontal layers. And a list of communication data volume between each two nodes that don't belong to the same bus needs to be provided. The total communication data volume V can be calculated using equation (14). V_i represents a communication data volume in the list.

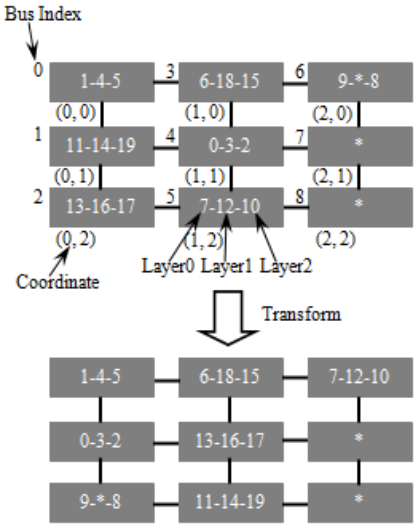


Fig. 7. Transformation of the mapping result of the optimal balanced distribution of traffic

$$V = \sum_{i=0}^{n} V_i \times \left(\left| X_{i0} - X_{i1} \right| + \left| Y_{i0} - Y_{i1} \right| \right) \quad (14)$$

Move the last bus to the next position first and calculate the total communication data volume every time. If the last bus has no position to move, then move the penultimate one. Until the first bus move to the last position, the transformations are finished. The transformation of which the total communication data volume is the least is that we need.

IV. Mapping Results Analysis

In the experiment of this paper, we found two optimal mapping results, optimal energy consumption and optimal balanced distribution of traffic, based on the task graph in Fig. 2. Unite the mapping results of stacks and the mapping results of horizontal layers, the final mapping results of 3D NoC-Bus hybrid architecture can be derived as shown in Fig. 8.

Assuming D is the Manhattan distance between two nodes in the 3D NoC-Bus hybrid architecture.

$$D = D_{horizontal} + D_{vertical} \quad (15)$$

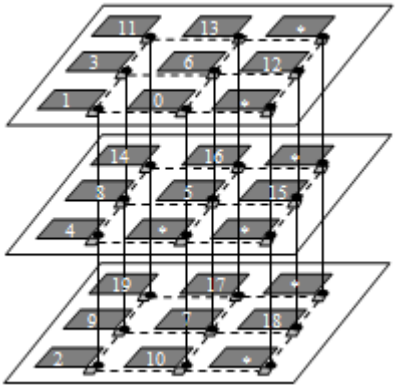
So the $D_{horizontal}$ of node 11 and node 8 of the optimal balanced distribution of traffic in Fig. 8 is 3 and the $D_{vertical}$ is 2. Using the Manhattan distance, we can calculate the energy consumption during communication by the following equation.

$$\begin{cases} E_{link} = E_{horizontal} \times D_{horizontal} + E_{vertical} \times D_{vertical} & (16) \\ E_{horizontal} = 1/2 \times \alpha \times C_{horizontal} \times V_{dd}^2 & (17) \end{cases}$$

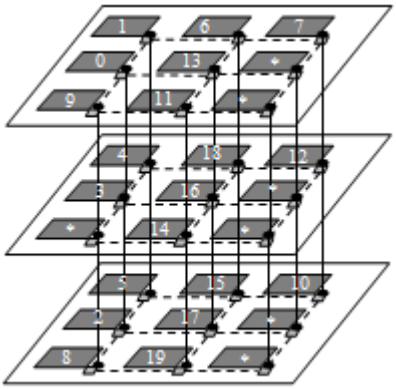
$E_{horizontal}$ is determined by the physical capacitance of horizontal link $C_{horizontal}$, supply voltage V_{dd} and activity factor α of the network communication [1]. $E_{vertical}$ can be calculated in a similar manner like $E_{horizontal}$ in (6).

According to the technology mentioned in literature [1], bit

transmission energy can be respectively calculated as 0.127 pJ and 9.56 × 10^{-3} pJ of horizontal link and vertical link. With that, the total transmission energy of optimal energy consumption is 0.275 uJ, just 76.4% of the transmission energy of optimal balanced distribution of traffic, which is 0.36 uJ.



(a) Optimal Energy Consumption



(b) Optimal Balanced Distribution of Traffic

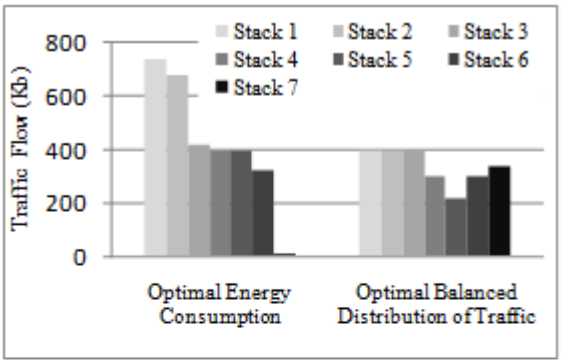
Fig. 8. Mapping results of the two optimal goals of 3D NoC-Bus hybrid architecture

Although optimal energy consumption can save much energy than optimal balanced distribution of traffic, but the traffic distribution has large gap between stacks and layers, which is not conducive to system heat dissipation. The difference of traffic distribution between optimal ways is obvious, as shown in Fig. 9.

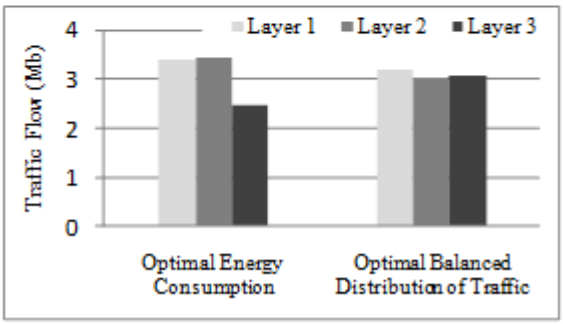
No matter among stacks or layers, it is obvious that the traffic distribution is more uniform in optimal balanced distribution of traffic than optimal energy consumption. The sum of variances of the stacks and layers' communication data volume of the optimal balanced distribution of traffic is only 5% of the optimal energy consumption.

The two optimal ways tested here are two extreme ways, one of which only considering reducing transmission energy, and the other is just for the sake of heat dissipation. Considering both of them by uniting F_{stacks} in (2) and S_v in (3), the unified equations can be represented by (7).

$$\begin{cases} F_{stacks} \geq F_{\min} \\ Min(S_v) \end{cases} \quad or \quad \begin{cases} S_v \leq S_{\max} \\ Max(F_{stacks}) \end{cases} \tag{18}$$



(a) Traffic distribution of buses of the two optimal goals in R1



(b) Traffic distribution of layers of the two optimal goals in R1

Fig. 9. Traffic distribution of the two optimal goals of 3D NoC-Bus hybrid architecture

F_{min} and S_{max} are set according to the results of bus combinations. On the basis of bus combinations in the step of mapping to stacks, we set F_{min} = 12 Mb and the map of optimal balanced distribution of traffic can be gotten as shown in Fig. 10.

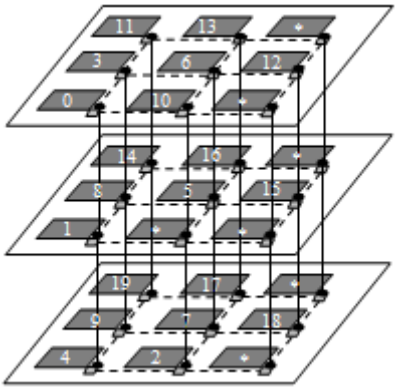


Fig. 10. Map of optimal balanced distribution of traffic under $F_{stacks} \geqslant 12$Mb

Using (6), the total transmission energy in Fig. 10 is 0.33 uJ, accounting for 91.7% and 120% of optimal balanced distribution of traffic and optimal energy consumption. And the sum of variances of the stacks and layers' communication data volume is between the variances of the two optimal goals, too.

By setting some other constraints, the proposed method can be used further. Such as, find the relationship between the stack's temperature and communication data volume. Define a threshold temperature of stacks, and we can find the most energy saving solution by using the proposed method under the circumstance that each stack's temperature is less than the threshold temperature.

In addition, three applications including VOPD, MPEG4, and one randomly generated application R1 are tested to check the improvement achieved by the method. The application information and the size of relevant 3D NoC-Bus Mesh architecture are listed Table II.

As is described in the problem formulation, traffic is the input we know and it is used to measure the heat. In order to show the potential energy reduction the proposed method can achieve under the same optimal goal, the method without the third step is employed as the baseline algorithm.

TABLE II
MAPPING PARAMETERS IN DIFFERENT APPLICATIONS

Application	IP cores	Flows	Valid buses	Total traffic	Size of NoC
VOPD	12	15	6	3494	3×2×2
MPEG4	12	26	6	3466	3×2×2
R1	20	27	7	4720	3×3×3

By adding the condition that α=0.8, we can get three optimal results in condition (7) under α=1, optimal energy consumption OEC, optimal OEC without the third step OECW, and OEC under α=0.8 OECU_0.8 in Fig. 11.

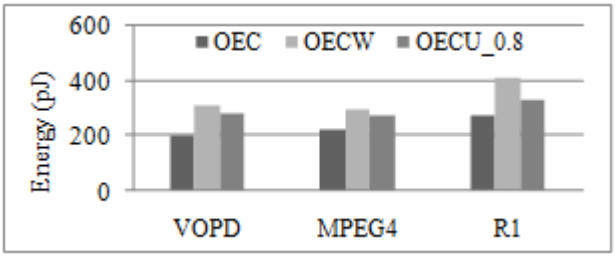


Fig. 11. Energy consumption in different optimal conditions

V. CONCLUSIONS

In this paper, we propose a method to find a reasonable map of traffic equilibrium and energy minimization to the 3D NoC-Bus mesh architecture. Through making full use of the architecture's properties, we first map IP cores on the buses and then exchange nodes' locations on the buses to reach a uniform traffic distribution among layers. At last, replace buses' locations to minimize the energy consumption of horizontal layers' communication. Experiment results show that the energy consumption is decreased and an appropriate balance can be made between energy consumption and traffic equilibrium.

REFERENCES

[1] Cheng Y, Zhang L, Han Y, et al. Thermal-constrained task allocation for interconnect energy reduction in 3D homogeneous mpsocs. Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, 2013, 21(2), pp: 239-249.

[2] Li, Feihui, et al. "Design and management of 3D chip multiprocessors using network-in-memory." ACM SIGARCH Computer Architecture News. Vol. 34. No. 2. IEEE Computer Society, 2006, pp: 130-141.

[3] Goplen, Brent, and Sachin Sapatnekar. "Thermal via placement in 3D ICs."Proceedings of the 2005 international symposium on Physical design. ACM, 2005, pp: 167-174.

[4] Wong, Eric, and Sung Kyu Lim. "3D floorplanning with thermal vias." Design, Automation and Test in Europe, 2006. DATE'06. Proceedings. Vol. 1. IEEE, 2006, pp: 1-6.

[5] Bakir, Muhannad S., et al. "3D heterogeneous integrated systems: Liquid cooling, power delivery, and implementation." Custom Integrated Circuits Conference, 2008. CICC 2008. IEEE. IEEE, 2008, pp: 663-670.

[6] Coskun A K, Ayala J L, Atienza D, et al. Dynamic thermal management in 3D multicore architectures. Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE'09. IEEE, 2009, pp: 1410-1415.

[7] Zhu C, Gu Z, Shang L, et al. Three-dimensional chip-multiprocessor run-time thermal management. Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, 2008, 27(8), pp: 1479-1492.

[8] Zhou X, Yang J, Xu Y, et al. Thermal-aware task scheduling for 3d multicore processors. Parallel and Distributed Systems, IEEE Transactions on, 2010, 21(1), pp: 60-71.

[9] Dick R P, Rhodes D L, Wolf W. TGFF: task graphs for free. Proceedings of the 6th international workshop on Hardware/software codesign. IEEE Computer Society, 1998, pp: 97-101.

[10] Zhang T, Wu N, Zhou F, et al. An Energy and Traffic Aware Mapping Method for 3-D NoC-Bus Hybrid Architecture. Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2014, WCECS 2014, 22-24 October, 2014, San Francisco, USA, pp: 53-57.

[11] Zhou X, Yang J, Xu Y, et al. Thermal-aware task scheduling for 3d multicore processors. Parallel and Distributed Systems, IEEE Transactions on, 2010, 21(1), pp: 60-71.

[12] Banerjee K, Souri S J, Kapur P, et al. 3-D ICs: A novel chip design for improving deep-submicrometer interconnect performance and systems-on-chip integration. Proceedings of the IEEE, 2001, 89(5), pp: 602-633.

[13] Xie Y, Loh G H, Black B, et al. Design space exploration for 3D architectures. ACM Journal on Emerging Technologies in Computing Systems (JETC), 2006, 2(2), pp: 65-103.