ClusCross: A New Topology for Silicon Interposer-Based Network-on-Chip

Hesam Shabani
Lehigh University
Bethlehem, PA, USA
hes318@lehigh.edu

Xiaochen Guo
Lehigh University
Bethlehem, PA, USA
xig515@lehigh.edu

ABSTRACT

The increasing number of cores challenges the scalability of chip multiprocessors. Recent studies proposed the idea of disintegration by partitioning a large chip into multiple smaller chips and using silicon interposer-based integration (2.5D) to connect these smaller chips. This method can improve yield, but as the number of small chips increases, the chip-to-chip communication becomes a performance bottleneck.

This paper proposes a new network topology, ClusCross, to improve network performance for multicore interconnection networks on silicon interposer-based systems. The key idea is to treat each small chip as a cluster and use cross-cluster long links to increase bisection width and decrease average hop count without increasing the number of ports in the routers. Synthetic traffic patterns and real applications are simulated on a cycle-accurate simulator. Network latency reduction and saturation throughput improvement are demonstrated as compared to previously proposed topologies. Two versions of the ClusCross topology are evaluated. One version of ClusCross has a 10% average latency reduction for coherence traffic as compared to the state-of-the-art network-on-interposer topology, the misaligned ButterDonut. The other version of ClusCross has a 7% and a 10% reduction in power consumption as compared to the FoldedTorus and the ButterDonut topologies, respectively.

CCS CONCEPTS

• **Hardware → Communication hardware, interfaces and storage: Networking hardware**.

KEYWORDS

Silicon interposer, die stacking, network-on-chip, topology

ACM Reference Format:
Hesam Shabani and Xiaochen Guo. 2019. ClusCross: A New Topology for Silicon Interposer-Based Network-on-Chip. In *International Symposium on Networks-on-Chip (NOCS '19), October 17–18, 2019, New York, NY, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3313231.3352363

NOCS '19, October 17–18, 2019, New York, NY, USA

ACM ISBN 978-1-4503-6700-4/19/10…$15.00
https://doi.org/10.1145/3313231.3352363

1 INTRODUCTION

As the number of transistors increases, more processor cores can be integrated into a Chip Multi-Processor (CMP) to boost the computation throughput. With the invention of High Bandwidth Memories (HBMs) [20], memory bandwidth can also be significantly improved by connecting multiple 3D-stacked DRAMs to processor chips through silicon interposers to satisfy the overall demands from the processors cores. Each processor core, however, might need to access multiple memory locations and the increased number of cores also escalate coherence traffic among the cores. The on-chip networks are facing fundamental challenges to enable the scalability of the CMPs and to satisfy both the coherence and memory traffic demands. In the meanwhile, with the increasing number of cores, on-chip power consumption is about to exceed the total power budget due to the limitation of the power delivery network and thermal dissipation capability. On-chip network designs have to be power efficient to meet the system power constraint.

Inspired by silicon interposer-based memory integration (*e.g.*, HBM) which is also referred to as 2.5D integration, recent studies [16] proposed the idea of disintegration by taking apart a large system into smaller parts by using the interposer-based integration to improve overall yield. This is because a smaller chip has fewer components and hence is less likely to catch defects. Having multiple smaller chips instead of a big chip also provides modularity and a defective small chip can be replaced at a lower cost when re-integrated through interposers. Nevertheless, as multiple smaller chips are integrated through the interposers, the amount of chip-to-chip communications are increased. Heavy traffic through the interposers can become a performance bottleneck [16]. Moreover, any processor core can also access different parts of the on-chip memories. Hence, the memory traffic also needs to pass across different chips through the interposers. Even though disintegration can improve the yield and reduce the fabrication cost, the interconnection network can become a performance bottleneck if it is not carefully designed to overcome the challenges posed by the interposer-based multi-chip systems.

Topology is one of the most important elements in interconnection network design, which has a direct influence on network performance. In interposer-based systems, memory traffic can compete with coherence traffic for bandwidth [16]. The network topology should be designed to reduce such contention by increasing the number of links and bandwidth on segments that are critical to both memory and coherence traffic.

In this work, a new interconnection network topology, ClusCross, is proposed for silicon interposer-based multi-chip systems. This topology is based on the idea of clustering. In order to decrease the network diameter and increase the cross-chip bandwidth, ClusCross

maps a cluster of routers onto each small chip and increases the number of cross-cluster long links. As a result, the proposed topology can increase path diversity and bisection bandwidth, which can help to reduce contentions between memory and coherence traffic. In addition, the use of cross-cluster long links can effectively reduce the number of hop counts for long-distance communication in both memory and coherence traffic.

The main contributions of this paper include:

- Two versions of ClusCross on-chip network topology are proposed with the aim of improving network performance in NoC-on-interposer systems through decreasing average hop count and increasing cross-chip bandwidth by leveraging long links.
- Performance and cost of the proposed ClusCross topologies are evaluated and compared against other existing topologies using synthetic memory and coherence traffic.
- System performance of ClusCross topologies is evaluated using the PARSEC suite traces that are appropriate for CMP assessment.

The rest of the paper is organized as follows: In Section 2, a brief overview of the interconnection networks based on silicon interposers is provided and related work for both conventional NoC topologies and topologies for silicon interposer systems are discussed. Section 3 presents the structure of the ClusCross and two versions of this topology. In Section 4, evaluation results are shown using both synthetic traffic patterns and real applications. The proposed topologies are compared against other topologies designed for interposer-based systems. Section 5 concludes the paper.

2 BACKGROUND AND RELATED WORK

2.1 Interposer-Based Interconnection Networks

Technology scaling does not benefit wires as much as it does to transistors [6]. On-chip communication becomes a bottleneck for both power consumption and performance. Three dimensional (3D) integration promises to bring processing elements and memory components physically close to each other to reduce wire distance and hence overcome the communication bottleneck. True 3D integration, however, requires through-silicon vias (TSVs), which is complicated to implement on processor dies and might introduce severe thermal issues and die yield reduction [16], [8]. As an alternative, individual chips can be connected to silicon interposer layer through micro-bumps. Hence, memory and processor chips can be connected through a layer of silicon interposers on a substrate die to increase memory bandwidth.

Since interposer integration does not need TSVs in the silicon interposer layer, higher die yield and additional routing capabilities are provided for the system [22]. In addition, interposer-based systems have lower manufacturing and R&D cost as compared to the true 3D integration [22]. Although the physical design of the interposer integration also has technology-related challenges such as thermal management and pin assignment [24], these challenges are solvable in near term [22]. Consequently, interposer-based systems are the most promising near-term solution for die-stacking integration. Several commercial products of interposer-based ICs are already on the market [18], [19]. For example, the HBM uses TSVs to integrate stacks of DRAM dies and connects the DRAM stacks to processor die using silicon interposers. Multiple processor chips can be connected through silicon interposers as well, which is used by [16] as a design method to improve yield. This previous work [16] shows that there is a trade-off between yield and performance when changing the chip size. Based on this study, partitioning a 64-core system into four smaller chips achieves the best performance and yield trade-off. Figure 1 shows an example of an interposer-based system, which consists of four 16-core processor chips, an interconnection network, as well as four HBM DRAM stacks placed on the left and right side of the processor dies.

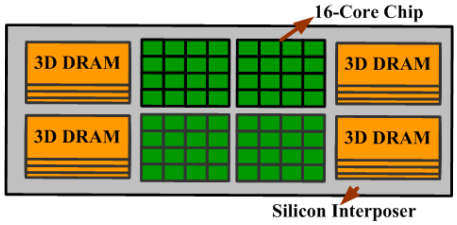

Figure 1: An illustration of a 64-core system composed of four 16-core processor chips and four HBM DRAMs.

The HBMs and processor dies are connected through a interposer layer. There are two types of the interposer layer: active and passive. An active interposer layer includes both interconnection links and routers, which requires transistors to be built on the interposer layer and hence can increase cost. A passive interposer layer [2], [7] has no transistors and has only interconnection links. A passive interposer layer tends to have a lower cost and higher yield as compared to the an active interposer layer due to the absence of transistors. When designing interconnection networks for interposer-based systems, it is important to use minimum number of transistors. Therefore, minimizing the number of routers and links on the interposer layer should be one of the design goals.

2.2 Conventional NoC Topologies

Network topology has a significant impact on communication latency because the number of the required hops for transmitting a message from a source to a destination node through routers can vary on different topologies. Moreover, the hop count influences network bandwidth and the power dissipation on interconnection networks. Typically, the smaller the average hop count is, the better the network performs. In addition, the bisection width has a determinant role in network throughput. Another factor to consider is whether the network can have deadlocks. It is better to have a deadlock-free routing algorithm because detecting and resolving deadlocks could introduce significant latency overheads [21].

Mesh and Torus topologies are commonly used in conventional network-on-chips because of their simplicity and regularity [23]. The main difference between these topologies is that there are additional long links in Torus networks for connecting the edge nodes. Leveraging these long links can improve load balance on the links and increase path diversity, which helps the communication paths to be quickly reconfigured to use alternative paths and can lead to have more efficient data transfer and link utilization [5]. Moreover,

adding long links can reduce network diameter, which is defined as the maximum distance between any two nodes. Nevertheless, implementing long links typically requires repeaters to be inserted to optimize energy and delay. Long links can consume more dynamic power in comparison with short links. Therefore, adding long links has a trade-off between performance and power.

Network topology also influence the complexity of router designs. The number of incoming and out-going links at each node determines the sizes of the input buffers and crossbar switches. Therefore, topologies with lower node degree is simpler to implement. In addition to the router complexity, the total number of links also directly influence the implementation cost and total power consumption. Mesh is cost-efficient in comparison with Torus because it has fewer links. FoldedTorus was proposed [4] to reduce the length of the long links in Torus, which improves performance. Increased the number of long links in FoldedTorus, however, leads to an increased area.

The design principle of the proposed ClusCross topology can be applied to both general-purpose on-chip networks and interposer-based systems. Section 3.2 discusses an example of general-purpose ClusCross, in which the maximum number of ports in the routers is kept the same as it is in Mesh, Torus, and FoldedTorus.

2.3 Silicon Interposer-Based Topologies

In a interposer-based system, the connection between processor die to the interposer layer is through micro-bumps, which has a 50μm pitch width. To accommodate this relatively large pitch width, the number of connections through micro-bumps is limited. Therefore, the network topology for interposer-based system typically uses concentrated nodes [1] to reduce routing nodes on the interposer layer [13]. Moreover, concentrated topologies reduce the average hop count to reach destinations for memory-bound requests. Prior work [13], [16] used a 4-to-1 concentration to design network topology for interposer-based system, which requires 8-port routers in the network.

Aligned and misaligned topologies are two types of concentrated topologies which have been proposed for interposer-based system with a minimally active area on the interposer layer. The key difference between aligned and misaligned topologies is that the misaligned topology places routers in between chips on the interposer layer, whereas the aligned topology only places links in between chips on the interposer layer. It is important to consider the cross-chip traffic in the interposer network because the cross-chip traffic includes both the coherence traffic and memory traffic. Unlike the coherence traffic within the chip, which can be transferred on chip or on the interposer, the cross-chip traffic has to be transferred on the interposer layer. Hence, the utilization of the middle links, which are also the bisection-crossing links, are important in interposer-based systems. In aligned topologies, the links that cross chips are shared between coherence and memory traffic. When both core-to-core coherence messages and memory messages want to pass through the cross-chip links in the interposer layer, messages queue up and then serialize behind each other to pass through the link. Whereas, in misaligned topologies, routers are the shared resources across chip, which allows both coherence and memory traffic to traverse through the routers at the same time. Therefore, the misaligned topologies are better at reducing queuing delays for massages to pass across chips [16].

In this work, we focus on the misaligned topologies because they tend to have better performance in comparison with aligned topologies [16]. The misalignment can be applied in X dimension (x) or X and Y dimensions (x+y) and it depends on the topology structure. FoldedTorus can be misaligned in both X- and Y- dimensions, whereas butterfly structures only can apply misalignment in the x dimension because adding one misaligned row to the butterfly topologies can change their structures. When misalignment is applied in one dimension or two dimensions, the total number of nodes in network can change as well. Accordingly, misalignment in both dimensions increases the total number of nodes and links in topology structure which is not cost and power efficient [16]. On the other hand, since minimizing active area in the interposer layer is preferred for NoC-on-interposer, topologies that have fewer number of nodes and links are preferred. In the meanwhile, it is desired to have topologies with smaller network diameter and lower average hop count. Therefore, topologies that misaligned to one dimension (x) are typically better for NoC-on-interposer systems.

Misaligned ButterDonut x and FoldedTorus x [16] have been proposed for NoC-on-interposer systems, which are demonstrated in Figure 2. ButterDonut x topology is designed based on the idea of increasing bisection width without the need to add more ports to routers and this topology adds more long links. The ButterDonut x topology has better performance as compared to the butterfly topologies evaluated in previous work [16] because it has more bisection width and fewer links. For example, ButterDonut x has twelve East-West bisection links, and eight North-South bisection links.

These topologies are subject to network deadlock because there are rings in their structure. Virtual channels [5] and bubble flow control [3] are two approaches which are widely used for avoiding deadlock. ButterDonut x topology uses flit-level bubble flow control with extra virtual channels in X-dimension to prevent deadlock from happening because the ButterDounut x topology only has rings in X-dimension [16].

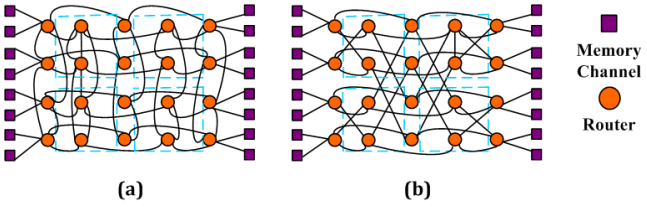

Figure 2: Illustrations of existing misaligned interposer-based topologies. (a) FoldedTorus x and (b) ButterDonut x.

Both ButterDonut x and FoldedTorus x use relatively short long links and many of these long links do not cross multiple chips. To use long links more efficiently, the proposed ClusCross topologies use fewer but longer long links to increase the bisection width.

3 CLUSCROSS TOPOLOGIES

This section presents the proposed ClusCross topology for both the interposer-based systems and as a general-purpose NoC network topology. The key idea of ClusCross is to map network clusters

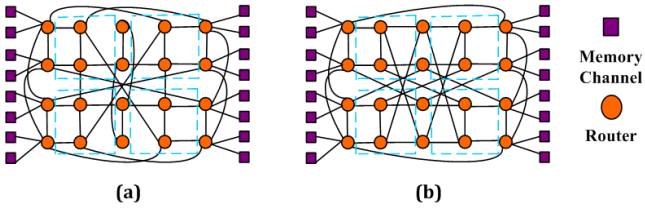

Figure 3: Illustrations of two versions of ClusCross topology. (a) ClusCross x-v1 and (b) ClusCross x-v2.

Table 1: A comparison of interposer-based topologies.

Topology (20 nodes)	Number of Links/Long links	Diameter	N-S/E-W Bisection Links	Avg. Hop
CMesh x	31/0	7	4/5	3.63
ButterDonut x	36/28	4	12/8	3.18
FoldedTorus x	40/22	4	8/10	3.18
ClusCross x-v1	38/14	4	10/12	3.03
ClusCross x-v2	40/16	4	12/14	2.98

onto disintegrated multi-core chips and use cross-cluster long links to increase cross-chip bandwidth. This topology can be generalized to other multi-core systems without interposers as well. Section 3.2 presents network characteristic comparison between the general-purpose ClusCross with other general-purpose topologies.

3.1 ClusCross Topologies Designed for Interposer-Based Systems

Two versions of misaligned ClusCross x are proposed based on the architecture that disintegrates 64-core to four 16-core chips and re-integrates them through silicon interposers. Both versions use long links to connect nodes across the chips and clusters, which can reduce network diameter and add more bisection bandwidth without increasing the number of ports in the routers. The structure of the two versions of ClusCross with 16 memory nodes are illustrated in Figure 3. The Figure shows long links without regular layout for easier illustration, but standard Manhattan routing is used for all wires in the evaluation. The design principle of these two versions of ClusCross is to start from a Mesh topology and replace some of the cross-cluster short links with cross-cluster long links. These topologies are designed to use only a few long links to increase bisection bandwidth and path diversity. Smaller average hop counts in these topologies can result in reduced network latency, which will be evaluated in Section 4.

The ClusCross x-v1 topology has fewer links, which has lower cost and power consumption as compared to the ClusCross x-v2 topology. ClusCross x-v2, however, increases the number of bisection width both vertically and horizontally, which leads to increased throughput saturation for memory traffic and coherence traffic. The number of East-West bisection links has influence on network throughput for memory traffic because the four HBMs are located on the left and right of the system. Adding East-West links to connect the chips on and left and right half of the system can reduce contention caused by memory traffic [16]. In addition, the total number of bisection links, East-West and North-South, have an impact on the network throughput for passing core-to-core coherence messages.

Table 2: A comparison of general-purpose topologies.

Topology (64 cores)	Number of Links/Long Links	Diameter	Bisection Links	Node Degree
Mesh	112/0	14	8	2-4
Torus	128/16	7	16	4
FoldedTorus	128/32	7	16	4
ClusCross	128/28	6	22	4

Table 1 presents important network parameters for the two versions of ClusCross x and three existing concentrated misaligned topologies. All five topologies have 20 concentrated router nodes on the interposer layer, four 16-core chips, and four HBMs with a total of 16 memory channels. Each node underneath the chips is connected to four cores, each node on the edge of the chip is connected to two cores and two memory channels, and each node between two chips is connected to four cores (two cores on each chip). CMesh x is a concentrated misaligned mesh topology, in which the 20 concentrated nodes are connected in a mesh topology. All four topologies with long links have smaller diameter as compared to CMesh x. The two versions of ClusCross have fewer long links as compared to ButterDonut x and FoldedTorus x, but the long links in ClusCross are longer as compared the the long links in ButterDonut x and FoldedTorus x. Hence, the average hop counts are reduced. The design of ClusCross intentionally uses cross-cluster long links. As a result, the N-S and E-W bisection links are increased.

ClusCross topologies are susceptible to network deadlock because of existing rings in their structure. Extra virtual channels [5] are applied for maintaining deadlock freedom in these topologies. Virtual channels are created through sharing a physical channel and using dedicated buffers for each source and destination pair. They are useful for improving network performance and saturation throughput because virtual channels allow alternative paths for transferring packets in the network and sharing network links [10]. Increasing the number of virtual channels, however, is not free. More virtual channels adds more hardware overhead.

3.2 General-Purpose ClusCross Topology

The design principle of ClusCross can be applied to general-purpose networks as well. This section describes the general-purpose version of ClusCross and compares the network parameters with other general-purpose topologies. However, the evaluation of the general-purpose ClusCross is not the focus of this work.

The ClusCross topology is useful for general-purpose network because it combines short links with long links and have important features from Mesh and Torus, which are widely adopted general-purpose on-chip interconnection network topologies. The general-purpose ClusCross uses similar idea of clustering and increases long link connections among clusters to reduce network diameter. Hence, it increases path diversity and bisection width for high-performance interconnection network design. Figure 4 shows an example of the general-purpose ClusCross topology. This topology is a symmetric design and all nodes have a degree of four, which is the same as Torus.

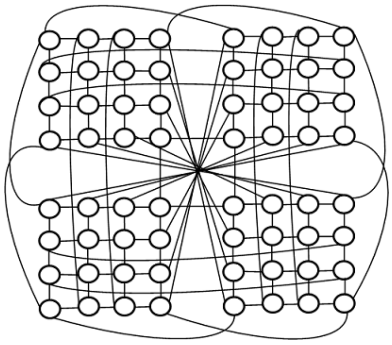

Figure 4: An illustration of a 64-node general-purpose ClusCross topology.

ClusCross tries to limit the use of the long links that are relatively longer. There are only four long links that are longer than the wrapped-around links in Torus and all the other long links are shorter than or the same as the wrapped-around links in Torus. The proposed general-purpose ClusCross also has fewer long links in comparison with the FoldedTorus. Like the Torus, the ClusCross topology is regular, all nodes have four links and also this topology is edge-symmetric, which can improve load balancing across the channels. Virtual channels can be applied to ClusCross to maintain deadlock freedom [9].

It is important to mention that both Torus and FoldedTorus require virtual channels to avoid deadlocks because using wrapped-around long links can create rings in the topology. A comparison on network characteristics between general-purpose ClusCross and three other general-purpose topologies is summarized in Table 2.

4 EVALUATION RESULTS

This section will focus on evaluating the proposed ClusCross topology for interposer-based system. This section will first present experimental setup, and then presents evaluation results on system performance, power, and area.

4.1 Experimental Setup

The proposed ClusCross topologies are evaluated and compared against two other topologies, ButterDonut x and FoldedTorus x, by using the BookSim 2.0 simulator [14] for system performance, area, and power consumption. It is feasible to use BookSim to simulate interposer-based systems because the back-end-of-line process in manufacturing metal layers of the interposer integration is the same as the process for metal interconnects in regular 2D chips. Metal density, resistance and capacitance of interposer's metal layers are considered the same as conventional on-chip wires [13]. BookSim 2.0 is a cycle-accurate interconnection network simulator that can provide system performance evaluation such as bandwidth and latency using synthetic traffic patterns and traces from real applications. BookSim 2.0 can also report power consumption and area. The models in BookSim 2.0 has been validated against the RTL implementations of the actual NoC router circuitry.

Injection mode and reply-and-request mode are two types of synthetic traffic patterns that can be used in BookSim to evaluate interconnection networks. In the injection mode, a specific injection rate of packets injected into the simulated network is used to measure the average latency and throughput. The performance measurement in the reply-and-request mode is typically based on the total time to finish the work and therefore this measurement is determined by the worst case. The reply-and-request mode is suitable to evaluate behavior of the memory traffic and there are limited number of Miss Status Handling Registers (MSHRs) for each node, which is defined as the maximum outstanding requests in the simulation. When all of MSHRs for a node are occupied, the next memory request will be stalled until one of the outstanding requests is replied [14], [17]. In this mode, the batch size defines a predetermined amount of work that each node needs to send before the simulation ends.

Synthetic traffic patterns in the simulator are designed for homogeneous systems. Misaligned topologies, however, have two different types of nodes: processing nodes and memory nodes which have different behaviors. Memory nodes only reply to requests and do not initiate any communication by themselves, whereas processing nodes can both reply to request or initiate communication by themselves. We modified BookSim to differentiate different behaviors of these two types of nodes and modeled memory traffic (core requests, memory replies) and coherence traffic (core to core). Network configuration parameters are listed in Table 3. The lengths of the long links are estimated to be the Manhattan distance between two nodes and are faithfully modeled for each topology in anynet configuration in BookSim to have a fair comparison. The latency of the link is proportional to the length of the link.

We used uniform random traffic pattern to emulate coherence traffic to evaluate ClusCross in comparison with other topologies, in which each source node is equal likely to send to any other destination nodes. Uniform random traffic distributes the packages uniformly, which is commonly used for network evaluation and it creates balanced loads [5].

The sample period in Table 3 is only applicable in injection rate mode. Latency and throughput of the network is determined after each sample period. After the warm up phase (three sample periods), measurement phase begins. After each sample period, statistics are reported. Before reporting final latency and throughput values, all of the measurement packets are drained from the network [15]. In Table 3, max samples define the maximum number of sample periods used in a simulation.

Figure 5 depicts how many virtual channels would be needed in each port for different topologies to maintain deadlock freedom and improve network performance. As shown in this Figure, saturation throughput will be constant after applying more than eight virtual channels. Hence, we chose eight virtual channels to minimize hardware overhead while providing a high saturation throughput.

In addition to synthetic traffic patterns, we also evaluated the proposed topology using realistic parallel applications. The traces are from simulations of PARSEC V2.1 on multiprocessor systems on Netrace 1.0 [12]. Netrace is a trace-based framework which generated packet traces by simulating a 64-core system through running the PARSEC V2.1 benchmark on M5 simulator [11]. Netrace considers dependencies between packets in the simulations. The architecture parameters used in Netrace are summarized in Table 4.

Table 3: Network Parameters.

Common Parameters	
Virtual Channels	8, 8 Flit router buffer size
Channel Width	128
Router Pipeline	4 Stages
Sample Period	10,000
Max Samples	10
Technology Node	32 nm
Clock Frequency	1 GHz
Batch Size	100,000
Outstanding Requests	4
Routing Algorithm	
CMesh x, FoldedTorus x	Dimension Order Routing
ButterDonut x, ClusCross x	Shortest-Path

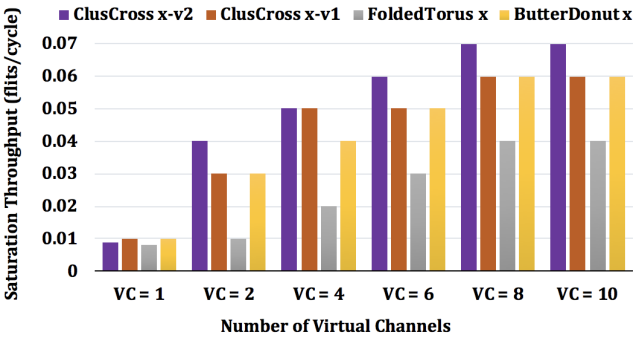

Figure 5: Saturation throughput of topologies for different numbers of VCs with the shortest-path routing algorithm under coherence traffic.

Table 4: Architecture Parameters.

Core	64 Cores, 2GHz, Alpha ISA, In-Order
L1 Cache	32KB Instruction/32KB Data, 4-Way Associative, 64B Lines, 3 Cycle Access Time
L2 Cache	64 Bank Fully Shared S-NUCA, 16MB, 64B Lines, 8-Way Associative, 8 Cycle Bank Access Time
Memory	150 Cycle Access Time, 8 On-Chip Memory Controllers
Coherence Protocol	MESI

PARSEC V2.1 includes a set of parallel applications, which are used in the evaluation. The simulated cycles, number of packets, average injection rate, and size of the header of each application are listed in Table 5.

Table 5: PARSEC V2.1 applications.

Benchmark	Simulated Cycles	Simulated Packets	Avg Injection Rate	Size of Header (B)
Blackscholes -Large	5833784581	113795888	0.019506	247
Bodytrack -Large	4577920449	385863891	0.084288	244
Canneal -Medium	23109826318	372046797	0.016099	243
Dedup -Medium	5450782575	431833996	0.079224	241
Ferret -Medium	8263033596	287425404	0.034784	242
Fluidanimate -Large	10172100050	187830527	0.018465	247
Swaptions -Large	1754464418	310331287	0.176881	244
Vips -Medium	5431630907	334870995	0.061652	240
X264 -Medium	42969045899	584828673	0.013610	240

4.2 System Performance Evaluation Using Injected Traffic Patterns

Coherence and memory traffic are emulated using different injected traffic patterns. Figure 6 presents the network latency of different topologies under different coherence traffic injection rates. As shown in this Figure, ClusCross x-v2 has the lowest average latency in comparison to the other topologies because of a smaller number of average hop counts in the network. For instance, ClusCross x-v2 has a 10.2% and a 12.9% average latency reduction at the 0.04 injection rate in comparison with ButterDonut x and FoldedTorus x, respectively. Additionally, this Figure shows that ClusCross x-v2 has the best saturation bandwidth as compared to the other topologies because it has the highest number of total bisection links, North-South and East-West, in the network, which has a significant influence on the saturation throughput for coherence traffic.

In order to analyze the amount of average packet latency for memory traffic, we evaluate performance by using both the injection mode and the reply-and-request mode in the simulator. Figure 7 shows the average packet latency and saturation throughput for memory nodes in injection mode. ClusCross x-v2 has a small reduction in average latency at the 0.07 injection rate in comparison to ButterDonut x and ClusCross x-v1, while has a 8.3% reduction as compared to FoldedTorus x. ClusCross x-v2 has a better saturation throughput as compared with the others because of having highest number of East-West bisection links, which is important for memory traffic. This means that the cross-cluster long links are particular useful for memory intensive workloads.

We also evaluated packet latency of all topologies using the reply-and-request mode to emulate memory traffic on the simulator. As we mentioned before, this mode simulates a specific batch size till all of the requests are replied, and the batch size is set in the simulator. The results are shown in Table 6. The two versions of ClusCross have lower average latency than the other two topologies do.

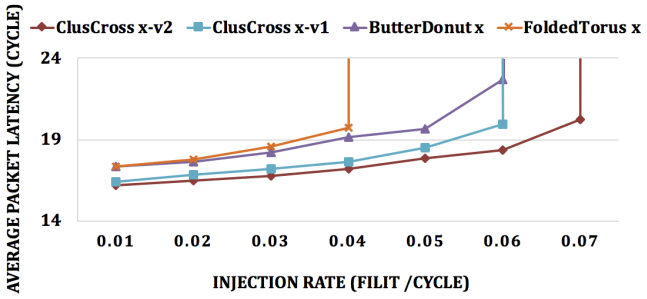

Figure 6: Average packet latency and saturation throughput of different network topologies for coherence traffic.

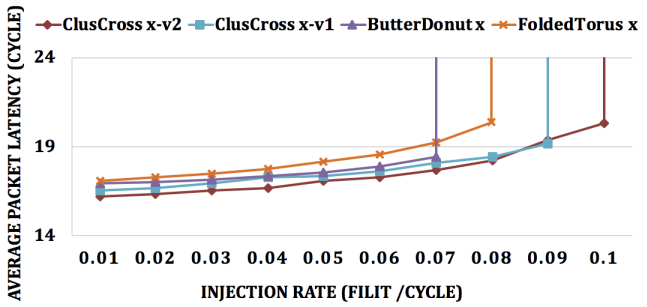

Figure 7: Average packet latency and saturation throughput of different network topologies for memory traffic.

Table 6: Average packet latency of memory traffic in reply-and-request batch mode.

ClusCross x-v2	ClusCross x-v1	ButterDonut x	FoldedTorus x
18.27	18.33	18.87	19.36

4.3 System Performance Evaluation Using PARSEC

In this section, the PARSEC benchmark suite is used to evaluate system performance of ClusCross in comparison with other topologies. Figure 8 shows total simulation runtime required to complete each PARSEC applications. These results are consistent with results of average packet latency in Figure 6. For application workloads with limited network pressure, topologies exhibit similar performance. Figure 9 presents average packet latency for different topologies normalized to CMesh x. Results have shown that ClusCross x-v2 has the lowest latency in comparison to the other topologies. ClusCross x-v1 also has lower latency as compared to FoldedTorus x and ButterDonut x.

4.4 Power and Area Evaluation

The power consumption in BookSim is calculated for 32 nm technology node. The simulator uses an analytical model to assess the area and power consumption. The size of the crossbar switch, number of connections, and number of buffers determine the area and static power consumption. Recorded activities in different components during simulation are applied to calculate the dynamic power consumption. Channel and switch components are the dominant factors

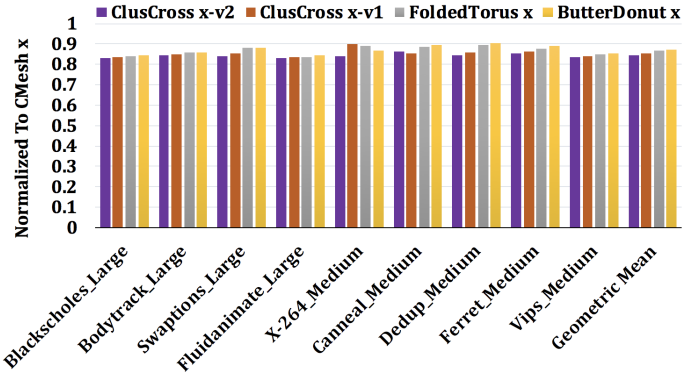

Figure 8: Total simulation runtime normalized to CMesh x.

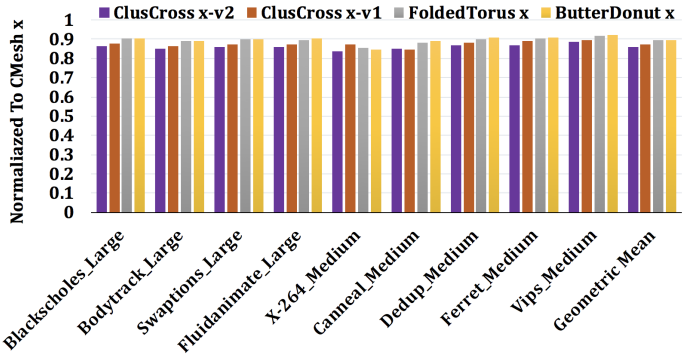

Figure 9: Average packet latency normalized to CMesh x.

which contribute the most to the total power consumption of the on-chip network. Figure 10 shows power consumption breakdown of different topologies. The main difference in power consumption in these topologies is the channel component. ButterDonut x consumes more power in comparison with the other topologies because it has more long links. Both versions of ClusCross consume less power on channels as compared to ButterDonut x and FoldedTorus x. This is because ClusCross uses fewer long links even though some of the long links are longer than the ones used in ButterDonut x and FoldedTorus x. The main reason for consuming more power in longer links is that longer wires are more resistive and have greater parasitic capacitance. After repeater insertion optimization, longer links still tend to consume more power than shorter links.

Figure 11 shows the area breakdown of on-chip network components for different topologies. The number of total links and length of the links determine the channel area, which is similar among different topologies. The main differences came from the switch component because all routers in ClusCross x-v2 and FoldedTorus x have eight ports, but some of routers in the two other topologies have fewer ports. Therefore, it is expected that the topologies which have routers with fewer ports have a smaller area.

5 CONCLUSION

This paper proposes a new class of on-chip network topology, ClusCross, designed to improve network performance parameters for NoC-on-interposer systems. The key idea is to map network

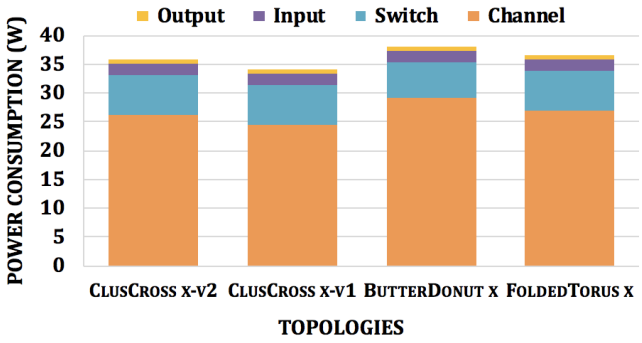

Figure 10: Power consumption breakdown of topologies.

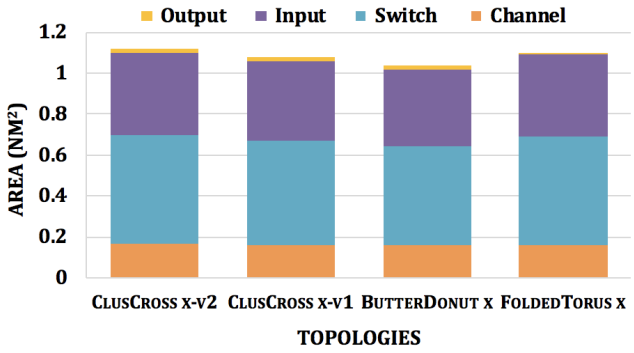

Figure 11: Area breakdown of different topologies.

clusters on to disintegrated chips and use cross-cluster long links to increase cross-chip bandwidth and decrease average hop count without changing the number of ports in the routers. Two versions of ClusCross are presented and evaluated. Synthetic memory and coherence traffic are used to compare ClusCross with other NoC-on-interposer topologies. In addition, traces from PARSEC benchmarks are also used to evaluate runtime and average network latency. The results show that ClusCross offers performance improvements on saturation throughput and average packet latency for interposer-based NoC with lower power consumption and similar area overheads as compared to the state-of-the-art topologies for interposer-based systems. Generally, ClusCross x-v1 works better for applications that have a power constraint; while, ClusCross x-v2 works better for applications that require better throughput and faster communications, because it has more bisection bandwidth and lower latency.

6 ACKNOWLEDGMENT

The authors would like to thank the reviewers who provided helpful suggestions which have improved the manuscript. This material is based upon work partially supported by the National Science Foundation at Lehigh University under Grant CCF-1750826 and CCF-1723624. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] James Balfour and William J. Dally. 2014. Design Tradeoffs for Tiled CMP On-chip Networks. In *ACM International Conference on Supercomputing 25th Anniversary Volume*. ACM, New York, NY, USA, 390–401. https://doi.org/10.1145/2591635.2667187

[2] Bryan Black. 2013. Die stacking is happening. In *Intl. Symp. on Microarchitecture, Davis, CA*.

[3] Lizhong Chen and Timothy M Pinkston. 2013. Worm-bubble flow control. In *2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 366–377.

[4] William J. Dally and Brian Towles. 2001. Route packets, not wires: on-chip inteconnection networks. In *Proceedings of the 38th annual Design Automation Conference*. Acm, 684–689.

[5] William J. Dally and Brian Towles. 2004. *Principles and practices of interconnection networks*. Elsevier.

[6] Bijan Davari, Robert H. Dennard, and Ghavam G. Shahidi. 1995. CMOS scaling for high performance and low power-the next ten years. *Proc. IEEE* 83, 4 (April 1995), 595–606. https://doi.org/10.1109/5.371968

[7] Yangdong Deng and Wojciech P. Maly. 2001. Interconnect characteristics of 2.5-D system integration scheme. In *Proceedings of the 2001 international symposium on Physical design*. ACM, 171–175.

[8] Xiangyu Dong, Jishen Zhao, and Yuan Xie. 2010. Fabrication Cost Analysis and Cost-aware Design Space Exploration for 3-D ICs. *Trans. Comp.-Aided Des. Integ. Cir. Sys.* 29, 12 (Dec. 2010), 1959–1972. https://doi.org/10.1109/TCAD.2010.2062811

[9] Jose Duato, Sudhakar Yalamanchili, and Lionel Ni. 2002. Interconnection Networks: An Engineering Approach, M. Kaufmann Pub. *Inc., USA* (2002).

[10] Masoumeh Ebrahimi and Masoud Daneshtalab. 2017. EbDa: A New Theory on Design and Verification of Deadlock-free Interconnection Networks. *SIGARCH Comput. Archit. News* 45, 2 (June 2017), 703–715.

[11] Mark Gebhart, Joel Hestness, Ehsan Fatehi, Paul Gratz, and Stephen W. Keckler. 2009. *Running parsec 2.1 on m5. University of Texas at Austin.* Technical Report. Department of Computer Science, Technical Report# TR-09-32.

[12] Joel Hestness and Stephen W. Keckler. 2011. *Netrace: Dependency-tracking traces for efficient network-on-chip experimentation. The University of Texas at Austin, Dept. of Computer Science.* Technical Report. Tech. Rep.

[13] Natalie E. Jerger, Ajaykumar Kannan, Zimo Li, and Gabriel H. Loh. 2014. Noc architectures for silicon interposer systems: Why pay for more wires when you can get them (from your interposer) for free?. In *2014 47th Annual IEEE/ACM International Symposium on Microarchitecture*. IEEE, 458–470.

[14] Nan Jiang, Daniel Becker, George Michelogiannakis, James Balfour, Brian Towles, David E. Shaw, John Kim, and William J. Dally. 2013. A detailed and flexible cycle-accurate network-on-chip simulator. In *2013 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE, 86–96.

[15] Nan Jiang, George Michelogiannakis, Daniel Becker, Brian Towles, and William J. Dally. 2019. BookSim 2.0 User's Guide. (05 2019).

[16] Ajaykumar Kannan, Natalie E. Jerger, and Gabriel H. Loh. 2015. Enabling Interposer-based Disintegration of Multi-core Processors. In *Proceedings of the 48th International Symposium on Microarchitecture (MICRO-48)*. ACM, New York, NY, USA, 546–558. http://doi.acm.org/10.1145/2830772.2830808

[17] Hanjoon Kim, Seulki Heo, Junghoon Lee, Jaehyuk Huh, and John Kim. 2010. On-Chip Network Evaluation Framework. In *SC '10: Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*. 10–10. https://doi.org/10.1109/SC.2010.35

[18] Joe Macri. 2015. AMD's next generation GPU and high bandwidth memory architecture: FURY. 1–26. https://doi.org/10.1109/HOTCHIPS.2015.7477461

[19] Liam Madden, Suresh Ramalingam, Xin Wu, Ephrem Wu, Bahareh Banijamali, Namhoon Kim, and Khaldoon Abugharbieh. 2013. Xilinx stacked silicon interconnect technology delivers breakthrough FPGA performance. 40 (05 2013), 6–11.

[20] Mike O'Connor. 2014. Highlights of The High-Bandwidth Memory (HBM) Standard. In *Memory Forum Workshop*.

[21] Umit Y. Ogras and Radu Marculescu. 2006. " It's a small world after all": NoC performance optimization via long-range link insertion. *IEEE Transactions on very large scale integration (VLSI) systems* 14, 7 (2006), 693–706.

[22] Sergii Osmolovskyi and Jens Lienig. 2017. Physical design challenges and solutions for interposer-based 3D systems. In *Reliability by Design; 9. ITG/GMM/GI-Symposium*. VDE, 1–8.

[23] Akram Reza, Hamid Sarbazi-Azad, Ahmad Khademzadeh, Hesam Shabani, and Behrad Niazmand. 2014. A loss aware scalable topology for photonic on chip interconnection networks. *The Journal of Supercomputing* 68, 1 (2014), 106–135. https://doi.org/10.1007/s11227-013-1026-4

[24] Xiaowu Zhang, Jong Kai Lin, Sunil Wickramanayaka, Songbai Zhang, Roshan Weerasekera, Rahul Dutta, Ka Fai Chang, King-Jien Chui, Hong Yu Li, David Soon Wee Ho, Liang Ding, Guruprasad Katti, Suryanarayana Bhattacharya, and Dim-Lee Kwong. 2015. Heterogeneous 2.5D integration on through silicon interposer. *Applied Physics Reviews* 2, 2 (2015), 021308. https://doi.org/10.1063/1.4921463